



## PATENT ABSTRACTS OF JAPAN

(11) Publication number: **09305468 A**

(43) Date of publication of application: 28 . 11 . 97

(51) Int. CI

**G06F 12/00**  
**G06F 12/00**  
**G06F 9/44**  
**G06F 13/00**

(21) Application number: **08120498**

(22) Date of filing: 15 . 05 . 96

(71) Applicant: **HITACHI LTD**

(72) Inventor: **KOBAYASHI KEN  
YAMAMOTO YOICHI  
ASAMI MASATO**

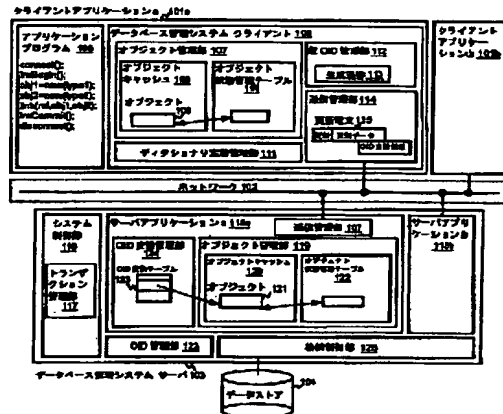
**(54) OBJECT MANAGING METHOD**

**(57) Abstract:**

**PROBLEM TO BE SOLVED:** To provide an object managing method which can efficiently process object operation by a client application without communicating with a server while guaranteeing object discrimination in an object-oriented data base system constituted in a client-server type.

**SOLUTION:** An object management part 107 of a database management(DBMS) client 106 allocates a temporary object identifier(OID), based upon a serial number (generation serial number 113) of object generation order, to a generated object 108. To an update message 115 that the database management(DBMS) client 106 generates, OID conversion information showing the position of the temporary OID in the message is added. A database management(DMMS) server 103 stores the generated object so that access efficiency of a data store 104 is improved, and converts the temporary OID into a format OID which is unique in the system according to the OID conversion information.

**COPYRIGHT: (C)1997,JPO**



(19)日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11)特許出願公開番号

特開平9-305468

(43)公開日 平成9年(1997)11月28日

(51)Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 3 3		G 0 6 F 12/00	5 3 3 J
	5 4 7			5 4 7 A
9/44	5 3 0		9/44	5 3 0 C
13/00	3 5 7		13/00	3 5 7 Z

審査請求 未請求 請求項の数5 O L (全 30 頁)

(21)出願番号 特願平8-120498

(22)出願日 平成8年(1996)5月15日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 小林 拳

神奈川県川崎市幸区鹿島田890番地の12

株式会社日立製作所情報・通信開発本部内

(72)発明者 山本 洋一

神奈川県川崎市幸区鹿島田890番地の12

株式会社日立製作所情報・通信開発本部内

(72)発明者 浅見 真人

神奈川県川崎市幸区鹿島田890番地の12

株式会社日立製作所情報・通信開発本部内

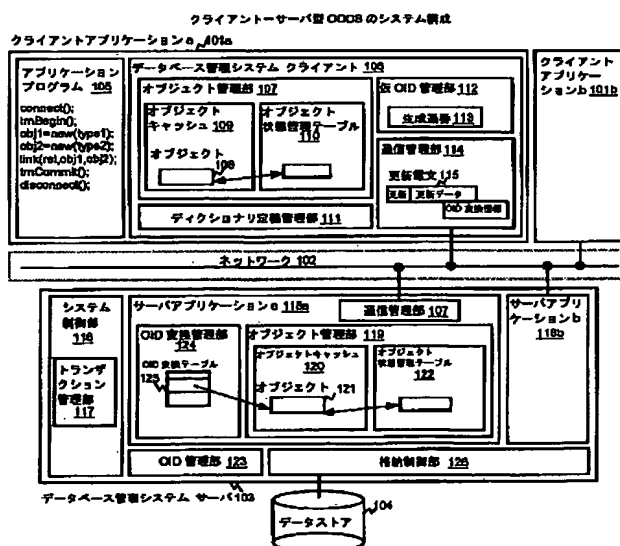
(74)代理人 弁理士 春日 譲

## (54)【発明の名称】 オブジェクト管理方法

## (57)【要約】

【課題】本発明の目的は、クライアントーサーバ型で構成されるオブジェクト指向データベースシステムにおいて、システム内でのオブジェクト識別性を保証しつつ、サーバと通信を行うことなく、クライアントアプリケーションでのオブジェクト操作を効率よく処理できるオブジェクト管理方法を提供するにある。

【解決手段】データベース管理システム (DBMS) クライアント106のオブジェクト管理部107は、オブジェクトの生成順の通し番号 (生成通番113) を基にした仮のオブジェクト識別子 (O I D) を、生成したオブジェクト108に割り付ける。データベース管理システム (DBMS) クライアント106が作成する更新電文115には、この電文中に含む仮O I Dの位置を示すO I D変換情報を付加する。データベース管理システム (DBMS) サーバ103は、生成したオブジェクトをデータストア104でアクセス効率が良くなるように格納し、O I D変換情報に基づき、仮O I Dをシステム内で一意となる正式なO I Dに変換する。



## 【特許請求の範囲】

【請求項1】 データベース管理システムクライアントは、クライアントアプリケーションで生成したオブジェクトを一時的な仮オブジェクト識別子で管理し、上記オブジェクトについて反映要求をされたデータベース管理システムサーバは、上記仮オブジェクト識別子をデータベースシステムで一意となる正式オブジェクト識別子に変換して、上記オブジェクトのデータストアへの格納を制御することを特徴とするオブジェクト管理方法。

【請求項2】 請求項1記載のオブジェクト管理方法において、上記データベース管理システムクライアントは、上記データベース管理システムサーバに対して上記オブジェクトについて反映要求をする際に、反映要求を指示する情報に、上記仮オブジェクト識別子から上記正式オブジェクト識別子に変換するための、補足的なオブジェクト識別子変換情報を付加することを特徴とするオブジェクト管理方法。

【請求項3】 請求項1記載のオブジェクト管理方法において、上記データベース管理システムクライアントは、上記オブジェクトを生成順の番号で管理し、この生成順の番号に基づいて上記仮オブジェクト識別子をそれぞれのオブジェクトに割り付けることを特徴とするオブジェクト管理方法。

【請求項4】 請求項1記載のオブジェクト管理方法において、上記データベース管理システムクライアントは、複数のオブジェクトを一括して上記データベース管理システムサーバに送信して、反映要求をすることを特徴とするオブジェクト管理方法。

【請求項5】 請求項4記載のオブジェクト管理方法において、上記複数のオブジェクト間にリンクを設定し、上記反映要求を指示する情報に、リンク先情報を設定する部分更新情報を付加することを特徴とするオブジェクト管理方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、オブジェクト管理方法に係り、特に、クライアントーサーバ型で構成されるオブジェクト指向データベースシステムに用いるに好適なオブジェクト管理方法に関する。

## 【0002】

【従来の技術】ネットワークを介したクライアントーサーバ型のシステム構成は、可用性や保守容易性を備えるための手段として、多くの計算機システムに適用されている。オブジェクト指向データベース(Object-Oriented Database:OODB)シス

テムにおいても、クライアントーサーバ型のシステム構成を適用することが一般的である。この場合、データベース管理システム(Database Management System:DBMS)をサーバとし、データベース管理システム(DBMS)で管理されるデータにアクセスするアプリケーションプログラム(Application Program:AP)をクライアントとしている。

【0003】オブジェクト指向データベース(OODB)システムで扱われるオブジェクトには、一般に「動的状態」と「永続状態」がある。オブジェクトの「動的状態」とは、通常、アプリケーションプログラム(AP)がオブジェクトに対して操作を行なえるように、計算機の主記憶上におかれた状態を指している。「動的状態」は、揮発的な性質を有している。主記憶の内容が失われるとき、例えば、アプリケーションプログラム(AP)プロセスの終了やデータベースシステムの停止などにより、「動的状態」は失われる。

【0004】「永続状態」とは、「動的状態」を再生するために退避された状態であり、通常、ディスクなどの二次記憶装置に保持される。なお、オブジェクトデータを「永続状態」から「動的状態」へ移すことを「活性化」(activate)という。また、「動的状態」でアプリケーションプログラム(AP)からの操作が可能な状態を、「活性化状態」という。

【0005】オブジェクトの「永続状態」を可能にする、即ち、オブジェクトの永続性を実現するための標準仕様として、オブジェクト・マネージメント・グループ社(Object Management Group Inc. (OMG))によって規定されたオブジェクト永続性サービス仕様(Object Persistence Service Specification (OPSS))が知られている。オブジェクト永続性サービス仕様(OPSS)は、オブジェクトの永続状態を管理するための機構とインタフェイスを与えるガイドラインである。例えば、オブジェクト永続性サービス仕様(OOPS)によれば、データストア(data store)とは、オブジェクトの永続状態のための記憶を与える機構をいい、永続識別子(Persistent Identifier:PID)とは、オブジェクトの永続状態の記憶場所を識別する情報を保持する値をいう。

【0006】複数のオブジェクトを動的および永続的に取り扱うときは、個々のオブジェクトをシステム内で一意に識別する機構が必要になる。オブジェクト・データベース・マネージメント・グループ社(Object Database Management Group (ODMG))が規定するオブジェクト指向データベース(OODB)の標準仕様のデータモデルによれば、

“The Object Database Stand

ard: ODMG-93 Release 1.1", Cattel, R. G. G. ed. Morgan Kaufmann Publishers, 1994に記載されているように、データベースシステム内の個々のオブジェクトを一意に識別するために、オブジェクト識別子 (Object Identifier: OID) が用いられる。

【0007】オブジェクト間の「関連」(relationship)は、オブジェクト指向データモデルにおける特長的なモデリングの手段である。「OMT」と呼ばれる一般的なオブジェクト設計技法によれば、「オブジェクト指向方法論 OMTモデル化と設計」, J. Rumbaugh他、羽生田栄一監訳、トッパン(1992年)に記載されているように、「関連」とは、オブジェクトの関係を確立するための手段であり、「リンク」(link)とは、オブジェクト間の物理的あるいは概念的な結合である。「関連」は、同一の意味を持つ「リンク」の集まり(定義)である。

【0008】一般的な関連(リンク)機能の実装方法としては、オブジェクト識別子(OID)をオブジェクトデータ内部に保持する方法がある。データ内部に保持しているオブジェクト識別子(OID)に一意に対応するオブジェクトを、関連付けられたリンク先オブジェクトと解釈する。

【0009】関連は本質的に双方向性を持つ。例えば、「人」と「会社」の間に関連「勤務する」を定義する場合、「勤務する」の逆関連「雇う」は「会社」と「人」との間の関連であり、どちらの方向をたどることも等しく意味を持ち、根底にある同一の関連を示す。

【0010】また、関連には、多対多などの「多重度」の概念があり、いくつかのオブジェクトがリンクで関連付けられるかが示される。例えば、「会社」が複数の「従業員」を「雇う」場合、この関連の多重度は1対多となる。

【0011】従来のオブジェクト指向データベース(OODB)システムでは、オブジェクト識別子(OID)がシステム内で一意となることを保証する。そのため、オブジェクト識別子(OID)の発行管理は、オブジェクト指向データベース(OODB)システム全体を管理するデータベース管理システム(DBMS)サーバで行なっている。

【0012】従って、クライアントアプリケーションで生成したオブジェクトにオブジェクト識別子(OID)を割り付けるには、データベース管理システム(DBMS)サーバとの通信が必要になる。

【0013】一方、オブジェクト間のリンクを管理するときには、一般にリンク先の情報として、オブジェクト識別子(OID)を用いる。そのため、新規オブジェクトにリンクを設定する際には、先にオブジェクト生成をデータベース管理システム(DBMS)サーバに伝えて

オブジェクト識別子(OID)を割り付けてから、リンク設定処理を行なわなければならない。

【0014】

【発明が解決しようとする課題】この一連の処理を簡潔に行なう方式としては、まずオブジェクト生成をデータベース管理システム(DBMS)サーバと通信してオブジェクト識別子(OID)を発行し、発行したオブジェクト識別子(OID)を受け取ってクライアントでリンク設定を行ない、リンク設定に伴う更新要求について別途データベース管理システム(DBMS)サーバと通信することが考えられる。

【0015】しかし、このようにして、リンクを管理する方式では、通信回数が増加するという問題がある。通信は処理コストが高いため、上記のような処理方式では、クライアントーサーバ型システムにおいて、クライアントの要求を効率良く処理することができず、高性能で動作することができない。

【0016】オブジェクト間のリンクを効率良く扱うクライアントーサーバ型オブジェクト指向データベース(OODB)システムの例として、例えば、オブジェクトデザイン(Object Design)社の製品として、オブジェクト・ストア(Object Store)が知られている。オブジェクト・ストアでは、オブジェクトの識別については、メモリ上のポインタを用いており、オブジェクト間の関連については、リンク先をポインタで直接的に表わしている。このポインタを永続的に使用できるように、オペレーティングシステム(OS)のページングおよび仮想記憶の機構を応用した「仮想記憶マッピングアーキテクチャ」と呼ばれる方法を採用している。

【0017】この方法では、クライアントでローカルにデータページ中のオブジェクトの格納構成が決定され、サーバではそのページの状態がそのままデータストアに格納される。従って、それぞれのクライアント毎に扱うオブジェクトをデータストア中で物理的に近い位置にまとめて格納(クラスタリング)することで、個々のクライアントの処理でのアクセス効率の向上が図れる。

【0018】しかし、このような格納構成では、個々のクライアントの処理とは無関係にオブジェクト操作を行なうような場合、例えば、データベース全体を対象にして、あるデータ構造を持つ(あるタイプに属する)オブジェクトを全件検索する場合、物理的に離れた位置のデータページへのアクセスが多く、特に、検索のためのインデクスなど特殊な機構を用いていない場合は、すべてのデータページを検索対象にしなければならず、効率が悪いものである。

【0019】「仮想記憶マッピングアーキテクチャ」による方法では、複数のクライアントが個別に扱うオブジェクト群をまとめて格納制御することは困難である。

【0020】本発明の目的は、クライアントーサーバ型

10

20

30

40

50

で構成されるオブジェクト指向データベースシステムにおいて、システム内でのオブジェクト識別性を保証しつつ、サーバと通信を行うことなく、クライアントアプリケーションでのオブジェクト操作を効率よく処理できるオブジェクト管理方法を提供するにある。

#### 【0021】

【課題を解決するための手段】上記の目的を達するため、本発明は、データベース管理システムクライアントは、クライアントアプリケーションで生成したオブジェクトを一時的な仮オブジェクト識別子で管理し、上記オブジェクトについて反映要求をされたデータベース管理システムサーバは、上記仮オブジェクト識別子をデータベースシステムで一意となる正式オブジェクト識別子に変換して、上記オブジェクトのデータストアへの格納を制御するようにしたものであり、かかる方法により、クライアントアプリケーションでのオブジェクト操作を、サーバとの通信を行うことなく、処理し得るものとなる。

【0022】上記オブジェクト管理方法において、好ましくは、上記データベース管理システムクライアントは、上記データベース管理システムサーバに対して上記オブジェクトについて反映要求をする際に、反映要求を指示する情報に、上記仮オブジェクト識別子から上記正式オブジェクト識別子に変換するための、補足的なオブジェクト識別子変換情報を付加するようにしたものであり、かかる方法により、正式オブジェクト識別子への変換を効率よく行い得るものとなる。

【0023】上記オブジェクト管理方法において、好ましくは、上記データベース管理システムクライアントは、上記オブジェクトを生成順の番号で管理し、この生成順の番号に基づいて上記仮オブジェクト識別子をそれぞれのオブジェクトに割り付けるようにしたものであり、かかる方法により、仮オブジェクト識別子の管理が容易となる。

【0024】上記オブジェクト管理方法において、好ましくは、上記データベース管理システムクライアントは、複数のオブジェクトを一括して上記データベース管理システムサーバに送信して、反映要求をするようにしたものであり、かかる方法により、クライアントサーバ間の通信回数を削減できるものとなる。

【0025】上記オブジェクト管理方法において、好ましくは、上記複数のオブジェクト間にリンクを設定し、上記反映要求を指示する情報に、リンク先情報を設定する部分更新情報を付加するようにしたものである。

#### 【0026】

【発明の実施の形態】以下、本発明の一実施形態によるオブジェクト管理方法について、図1乃至図19を用いて説明する。図1は、本発明の一実施形態によるオブジェクト管理方法を適用したオブジェクト指向データベース(OODB)システムの構成図である。

【0027】クライアントアプリケーション101a, 101bは、ネットワーク102を介して、データベース管理システム(DBMS)サーバ103と通信し、データストア104に格納されるオブジェクトにアクセスするクライアントサーバ型でシステムを構成している。

【0028】このシステムは、通信回数を増やすことなく、クライアントでのオブジェクト操作(特に、オブジェクト生成とオブジェクト間リンク設定)を行なえるように構成されている。そして、通信を回避するための基本的な処理方式として、以下の(1)~(3)を備えている。

【0029】(1)仮オブジェクト識別子(仮OID)の発行

クライアントアプリケーション101a, 101bは、データベース管理システム(DBMS)サーバ103と通信せずに、新規生成したオブジェクトに対して、仮のオブジェクト識別子(OID)を発行し、割り付ける。ここでは、この仮のオブジェクト識別子(OID)のことを「仮OID」と呼ぶことにする。

【0030】この「仮OID」は、データベース管理システム(DBMS)サーバ103がシステム内で一意になるように正式に発行するオブジェクト識別子(OID)である「正式OID」と対比して用いるものである。

【0031】仮OIDは、生成順の通番を、クライアントアプリケーション101a, 101bが、オブジェクト識別子(OID)の内部データに取り込んで作成する。

【0032】クライアントアプリケーション101a, 101bにおいて、オブジェクト操作対象となったオブジェクトを識別する値および操作要求の内容は、仮OIDを用いて記録される。

【0033】(2)更新反映情報の一括送信と、OID変換情報の付加

クライアントアプリケーション101a, 101bにおけるオブジェクト操作に対応するデータベース管理システム(DBMS)サーバ103へ反映要求は、トランザクションコミットを契機に一括して送信する。

【0034】反映要求の通信電文には、仮OIDを正式OIDに効率良く変換するための補足的な情報(OID変換情報)を付加しておく。

【0035】(3)オブジェクト識別子(OID)を変換してデータストアへ反映

データベース管理システム(DBMS)サーバ103は、上記のOID変換情報に従い、オブジェクトの生成通番を元に、仮OIDを効率良く正式OIDに変換し、通信電文中の仮OID中のオブジェクト操作要求をデータストアに反映する。

【0036】データストアへの格納は、データベース管

理システム (DBMS) サーバ103が制御し、クラスタリングなどによりアクセス効率の向上を図る。

【0037】次に、図1に示したクライアントアプリケーション101a、101b及びデータベース管理システム (DBMS) サーバ103の構成について説明する。

【0038】最初に、クライアントアプリケーション101a、101bの構成について説明する。クライアントアプリケーション101のアプリケーションプログラム (AP) 105は、データベース (DB) にアクセスする手順を指示する。アプリケーションプログラム (AP) 105は、データベース管理システム (DBMS) クライアント106が提供するアプリケーションプログラムインタフェース (Application Program Interface: API) の関数群を用いて記述される。クライアントアプリケーション105には、データベース管理システム (DBMS) クライアント106がバインドされる。

【0039】データベース管理システム (DBMS) クライアント106のオブジェクト管理部107は、クライアントアプリケーション101が扱うオブジェクト108を管理する。オブジェクト108は、オブジェクト管理部107のクライアントオブジェクトキャッシュ109上に保持される。

【0040】オブジェクト管理部107の中のオブジェクト状態管理テーブル110は、クライアントオブジェクトキャッシュ109上のオブジェクト108の状態を管理する。オブジェクト状態管理テーブル110のエントリは、各オブジェクト108に、1つずつ対応付けられる。

【0041】データベース管理システム (DBMS) クライアント106の中のディクショナリ定義管理部111は、オブジェクト108のデータ構造 (タイプ) およびアクセス方法の定義の集まりであるディクショナリ情報を管理する。アプリケーションプログラム105の動作に必要なディクショナリ定義情報が、クライアントアプリケーション101にバインドされるものとする。

【0042】データベース管理システム (DBMS) クライアント106の中の仮OID管理部112は、クライアントアプリケーション101で生成する個々のオブジェクト108を識別する仮OIDの発行を管理する。仮OIDは、仮OID管理部112で管理される生成順による生成通番113をもとに作成される。

【0043】データベース管理システム (DBMS) クライアント106の通信管理部114では、データベース管理システム (DBMS) サーバ103との通信を管理し、クライアントアプリケーション101におけるオブジェクト操作をデータベース (DB) に反映させるための更新電文115を管理する。

【0044】次に、データベース管理システム (DBM

S) サーバ103の構成について説明する。データベース管理システム (DBMS) サーバ103は、トランザクション管理やオブジェクト管理など、データベースシステムが提供する各機能を実現するためのプロセスの集まりから構成されている。

【0045】データベース管理システム (DBMS) サーバ103の中のシステム制御部116は、データベース (DB) システムを構成するプロセス群を監視し、システム全体の制御を行なう。

【0046】システム制御部116の中のトランザクション管理部117は、各クライアントアプリケーション101で実行されるトランザクションの管理を行なう。

【0047】サーバアプリケーション118a、118bは、クライアントアプリケーション101a、101bにそれぞれ1つずつ対応するプロセスで、クライアントアプリケーション101からの要求を受け付け、データベース管理システム (DBMS) サーバ103の各モジュール (管理部、制御部) に要求を伝える。

【0048】データベース管理システム (DBMS) サーバ103の中のオブジェクト管理部119は、サーバアプリケーション118のオブジェクト操作要求を受け付けて処理する。サーバオブジェクト管理部119は、クライアントアプリケーション101のオブジェクト108に対応するように、データベース管理システム (DBMS) サーバ103のオブジェクトキャッシュ120上でオブジェクト121を操作し、このオブジェクト121の状態は、オブジェクト管理情報テーブル122により管理する。

【0049】データベース管理システム (DBMS) 103の中のサーバOID管理部123は、データベース (DB) システムにおける正式OIDを発行し、一意性を保証するように管理する。

【0050】データベース管理システム (DBMS) 103の中のサーバOID変換管理部124は、クライアントアプリケーション101が発行した仮OIDと、正式OIDとの対応関係を管理する。対応関係は、サーバOID変換管理部124の中のOID変換テーブル125を用いて管理する。OID変換テーブル125の詳細については、図19を用いて後述する。

【0051】格納制御部126は、オブジェクトデータを格納するデータストア104とのデータ入出力を管理する。

【0052】データストア104は、オブジェクト121の永続状態のための記憶を与える機構である。

【0053】通信管理部107は、クライアントアプリケーション101からメッセージを受け取るバッファの管理を行う。

【0054】次に、図2を用いて、本実施形態によるオブジェクト管理方法の処理方式について説明する。図2は、本発明の一実施形態によるオブジェクト管理方法に

10

20

30

40

50

におけるクライアントアプリケーションでのアプリケーションプログラムの処理フローの概念を示すフローチャートである。なお、詳細な処理フローについては、図3以下を用いて後述する。

【0055】本実施形態においては、例えば、2つのオブジェクトを作成し、それらのオブジェクト間にリンクを設定するオブジェクト操作のプログラムの処理について説明する。

【0056】ステップ201において、データベース管理システム(DBMS)クライアント106は、データベース(DB)接続するためのデータベース管理システム(DBMS)クライアント106のアプリケーションプログラムインタフェース(API)関数をコールし、クライアントアプリケーション101とデータベース管理システム(DBMS)サーバ103との通信を確立する。ステップ201の詳細については、図3を用いて後述する。

【0057】ステップ202において、データベース管理システム(DBMS)クライアント106は、トランザクションを開始するAPI関数をコールする。ステップ202の詳細については、図6を用いて後述する。この後、トランザクション開始から、次のトランザクションコミットまで間の処理について、データベース(DB)システムで整合性が保証される。

【0058】ステップ203において、データベース管理システム(DBMS)クライアント106は、オブジェクト108を生成するAPI関数をコールし、「オブジェクト1」を生成する。ステップ203の詳細については、図7、8を用いて後述する。

【0059】ここで、オブジェクト生成APIの入力として、生成するオブジェクトのタイプを指定する手段があるものとする。例えば、ディクショナリ定義管理部111に定義されたタイプ識別子を指定するものとする。タイプ識別子は、データベース(DB)システムの一般的なディクショナリ定義操作により与えられ、指定されたタイプ識別子に対応するタイプ定義に基づくデータ構造のオブジェクトが生成されるものとする。

【0060】ステップ204において、データベース管理システム(DBMS)クライアント106は、ステップ203と同様にして、「オブジェクト2」を生成する。ここで、「オブジェクト1」と「オブジェクト2」で指定するタイプは、同じであっても、異なってもよい。ただし、それぞれのタイプ間に1対1の双方向関連が定義されているものとする。

【0061】ステップ205において、データベース管理システム(DBMS)クライアント106は、オブジェクト間リンク設定API関数をコールし、「オブジェクト1」と「オブジェクト2」の間にリンクを設定する。ステップ205の詳細については、図9、10、11を用いて後述する。ここで、リンク設定APIの入力

として、ディクショナリに定義された関連の関連識別子を指定するものとする。この指定により、関連の定義情報に従い、オブジェクト間のリンクが設定されるものとする。

【0062】ステップ206において、データベース管理システム(DBMS)クライアント106は、トランザクションコミットするAPI関数をコールする。ステップ206の詳細については、図12を用いて後述する。これにより、ステップ203以降の処理を、データベース(DB)に反映させる。

【0063】ステップ207において、データベース(DB)接続を解除するAPI関数をコールし、クライアントアプリケーション101とデータベース管理システム(DBMS)サーバ103との通信を解除する。ステップ207を終えた後、アプリケーションプログラムを終了する。

【0064】図3を用いて、図2のステップ201に示したデータベース管理システム(DBMS)クライアントにおけるデータベース(DB)接続APIの処理の詳細について説明する。図3は、本発明の一実施形態によるオブジェクト管理方法におけるデータベース(DB)接続APIの処理手順を示すフローチャートである。

【0065】ステップ301において、データベース管理システム(DBMS)クライアント106のオブジェクト管理部107は、制御情報のデータ領域を確保し、初期化する。

【0066】ここで、データベース管理システム(DBMS)クライアント106の制御データ構造について、図4を用いて説明する。図4は、本発明の一実施形態によるオブジェクト管理方法において用いるデータベース管理システム(DBMS)クライアントで取り扱う制御データの構造の説明図である。

【0067】この制御情報の構成により、クライアントアプリケーション101で一意となる仮OIDの発行を管理し、トランザクションコミット時に、一括して更新反映要求を行なうための情報を保持する。

【0068】データベース管理システム(DBMS)クライアント106の機能のための制御情報として、クライアント管理制御ヘッダ401、オブジェクト管理制御ヘッダ402、仮OID管理制御ヘッダ403、通信管理制御ヘッダ404がある。これらの制御情報のアドレスは、クライアント管理制御ヘッダアドレスとして、データベース管理システム(DBMS)クライアントの各モジュールに対してグローバルな値に保持され、各モジュールがこれらの制御情報にアクセスする。

【0069】クライアント管理制御ヘッダ401は、クライアントアプリケーション101のプロセス全体を管理するために必要な情報(ステータス)を保持している。例えば、アプリケーションプロセスの状態(DB接続状態、トランザクション実行状態)を保持している。

10

20

30

40

50

【0070】オブジェクト管理制御ヘッダ402は、オブジェクト管理部107を制御する情報を保持している。オブジェクト管理制御ヘッダ402は、キャッシュ制御ヘッダ405のアドレス、およびオブジェクト状態管理制御ヘッダ406のアドレスを保持している。

【0071】キャッシュ制御ヘッダ405は、オブジェクトキャッシュ109を管理するために使用し、オブジェクト108のデータのキャッシングの管理に必要な情報を保持している。キャッシュ制御ヘッダ405は、例えば、オブジェクトキャッシュ109のキャッシュ領域の先頭アドレス、利用可能領域の先頭アドレスを保持している。

【0072】オブジェクト状態管理テーブル制御ヘッダ406は、オブジェクト状態管理テーブル110を管理する。オブジェクト状態管理テーブル110のエントリを割り付けるために、オブジェクト状態管理テーブル110のテーブル領域の先頭アドレス、利用可能テーブルエントリの先頭アドレスなどの情報を保持している。

【0073】仮OID管理制御ヘッダ403は、仮OIDを発行するための情報を管理する。仮OIDの元となる生成順通番カウンタ113を保持している。

【0074】通信管理制御ヘッダ404は、通信管理情報を制御するために用い、通信電文を保持する領域である通信バッファ408の先頭アドレスなどを保持する。

【0075】本実施形態のシステムでは、クライアントアプリケーション101で行なったオブジェクト操作のデータベース管理システム(DBMS)サーバ103に対する反映は、コミット時に一括して行なう。反映する要求の内容を、データベース管理システム(DBMS)クライアント106で一時的に更新情報として保持し、オブジェクト状態管理エントリ407からポイントしておく。反映が必要な更新要求情報を効率良くに得るために、反映が必要なオブジェクト状態管理エントリ407を、オブジェクト状態管理制御ヘッダ402からポイントし、さらに、複数の更新対象のオブジェクト状態管理エントリ407を得られるように各エントリをポインタで連結しておく。ここでは、この連結のことを「反映要求チェーン409」と称する。

【0076】次に、ステップ302において、オブジェクト管理部107は、仮OID管理部112を初期化し、仮OID管理部112は、生成通番113を初期化する。

【0077】ステップ303において、オブジェクト管理部107は、通信管理部114を初期化し、以降のデータベース管理システム(DBMS)サーバ103への通信要求に備える。

【0078】ステップ304において、データベース管理システム(DBMS)クライアント106は、データベース(DB)接続要求電文を作成する。

【0079】ここで、本実施形態において用いる通信電

文のデータ構造について、図5を用いて説明する。図5は、本発明の一実施形態によるオブジェクト管理方法においてクライアントアプリケーションが作成する通信電文のデータ構造の説明図である。

【0080】通信電文501は、電文制御ヘッダ502と、複数(n個)のコマンドパラメタブロック503から構成されている。電文制御ヘッダ502は、通信電文501を制御するために必要な以下の情報を保持している。

10 【0081】(1)「電文の形式」(サーバへの送信データであることを示す値など)

(2)「電文のサイズ」

(3)「先頭ブロックオフセット」(先頭のコマンドパラメタブロック503の位置を示すオフセット値;通信電文501の先頭からのオフセット値として与えられている)

(4)「ブロック総数(n)」(電文501に保持するコマンドパラメタブロック503の総数)

20 コマンドパラメタブロック503は、ブロック制御ヘッダ504と、ブロックデータ504から構成されている。ブロック制御ヘッダ504は、各コマンドパラメタブロック503を制御する情報を有している。ブロック制御ヘッダ504は、例えば、「ブロック形式」、「ブロックサイズ」(ブロックのデータ長)、「コマンド識別子」(データベース(DB)接続要求であることなどを示す値など)、「ブロックデータオフセット」(ブロックデータの位置を示すオフセット値;コマンドパラメタブロック503の先頭からのオフセット値)、「次ブロックのオフセット」(次のブロックの位置を示すオフセット値;通信電文501の先頭からのオフセット値)を保持している。

【0082】以上説明した構造を用いることにより、例えば、データベース(DB)接続要求の電文メッセージでは、電文制御ヘッダ502内の「ブロック総数」を「1」とし、最初のブロック1のブロック制御ヘッダ504内の「コマンド識別子」に「DB接続」を示す値を設定するようにしている。

【0083】ステップ305において、通信管理部114は、データベース管理システム(DBMS)サーバ103へ通信電文501を送信する。ステップ305を終えた後、データベース(DB)接続処理を終了する。

【0084】ここで、プロセス間通信の手段および形態については、本実施形態の方法を実装するために特別な機能を必要とせず、周知の機構を用いてよい。

【0085】図6を用いて、図2のステップ202に示したデータベース管理システム(DBMS)クライアントにおけるトランザクション開始APIの処理の詳細について説明する。図6は、本発明の一実施形態によるオブジェクト管理方法におけるトランザクション開始API関数コールの処理手順を示すフローチャートである。



【0086】ステップ601において、データベース管理システム（DBMS）クライアント106は、オブジェクト管理部107にリセット要求し、オブジェクト管理部107がリセットする（制御情報を初期状態に戻し）ことにより、後のオブジェクト操作要求に備える。

【0087】ステップ602において、オブジェクト管理部107は、仮OID管理部112にリセット要求し、仮OID管理部112がリセットすることにより、後のオブジェクト生成時の仮OID発行に備える。

【0088】ステップ603において、データベース管理システム（DBMS）クライアント106は、トランザクション開始要求の通信電文を作成する。通信電文は、通信管理部114の通信バッファ領域に作成される。

【0089】ステップ604において、通信管理部114は、作成した通信電文を、データベース管理システム（DBMS）サーバ103へ送信する。データベース管理システム（DBMS）サーバ103は、トランザクション管理部117に、トランザクション開始要求が伝えられ、クライアントアプリケーション101のプロセスでのトランザクションが認識され、管理対象となる。

【0090】ステップ604を終えた後、この処理を終了する。

【0091】次に、図7及び図8を用いて、図2のステップ203に示したデータベース管理システム（DBMS）クライアントにおけるオブジェクト生成APIの処理の詳細について説明する。図7は、本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト生成APIの処理手順を示すフローチャートであり、図8は、生成されるオブジェクトのデータ構造の説明図である。

【0092】図7に示すステップ701において、オブジェクト管理部107は、ディクショナリに定義されているタイプ定義情報を参照する。参照するタイプ定義対象は、アプリケーションプログラム105からオブジェクト生成APIのパラメタで指定されており、このタイプ識別子に対応するタイプ定義情報を、ディクショナリ定義管理部111から取得する。

【0093】ステップ702において、オブジェクト管理部107は、オブジェクトキャッシュ109上にデータ領域を取得する。データ領域の大きさは、ステップ701で取得したタイプ定義情報を参照して、必要なサイズを算出する。

【0094】ステップ703において、オブジェクト管理部107は、ステップ701で取得したタイプ定義情報に従って、オブジェクト108を初期化する。オブジェクト108のデータ構造には、リンク先情報を保持する構造があり、このステップにおいて、リンク先情報を初期化する。

【0095】ステップ704において、オブジェクト管

理部107は、このオブジェクトの状態を管理するために、オブジェクト状態管理テーブル110のエントリを割り付ける。

【0096】ステップ705において、オブジェクト管理部107は、ステップ704で取得したオブジェクト状態管理テーブル110のエントリを初期化する。ここで、オブジェクト状態管理テーブル110のエントリの中のオブジェクト状態を保持する領域のフラグを、新規状態であることを示すONに設定する。

【0097】ステップ706において、オブジェクト管理部107は、このオブジェクト状態管理テーブル110のエントリを、反映要求チェーン409につなぐ。反映要求チェーン409は、次のエントリをポインタで指示する。

【0098】ステップ706を終えた後、オブジェクト生成API処理を終了する。

【0099】次に、図8を用いて、データベース管理システム（DBMS）クライアント106におけるオブジェクト108のデータ構造について説明する。オブジェクト108は、オブジェクト制御ヘッダ801と、オブジェクトデータ領域802から構成されている。オブジェクト制御ヘッダ801は、オブジェクトデータを制御情報するために、以下の情報から構成されている。

【0100】（1）「オブジェクト状態管理テーブル110のオブジェクト状態管理エントリ806のアドレス」

（2）「オブジェクトの種別」

（3）「オブジェクト全体のデータサイズ」

（4）「自分自身のOID」（このオブジェクトに割り付けられたOID）

クライアントアプリケーション101で生成された時点では、「自分自身のオブジェクト識別子（OID）」は設定されていない。仮OIDは、当該オブジェクト操作において、オブジェクト識別子（OID）が必要になるとき、例えば、リンク設定を行なうときに発行され、割り付けられる。リンク設定時の仮OIDの割り付けについては、図9を用いて後述する。

【0101】「自分自身のOID」の中は、「オブジェクト識別子（OID）の形式」（「仮」であるか「正式」であるかを示す値）と、一意性のための「識別値」から構成されている。

【0102】仮OID803の場合には、「OID形式」には、仮オブジェクト識別子（OID）形式であることを示す値が設定され、「識別値」には、割付対象オブジェクトの生成時の生成通番が設定される。

【0103】正式OID804の場合には、「OID形式」には、正式オブジェクト識別子（OID）形式であることを示す値が設定され、「識別値」には、データベース管理システム（DBMS）サーバ103のOID変換管理部124において、データベース（DB）システ

ム内で一意になるように作成された値が設定される。ここでは、オブジェクトデータ格納位置情報に基づく永続識別子PIDを識別値として用いる。

【0104】オブジェクトデータ領域802には、オブジェクトの内容を示す「オブジェクトデータ」を保持している。この領域に、リンク先情報805も保持される。リンク先情報805には、「リンク先OID」（リンク先オブジェクトのオブジェクト識別子（OID））が保持される。リンク先オブジェクトが生成されてトランザクションコミットする前ならば、仮OID803を保持することになる。リンク先オブジェクトがデータストアに格納されている既存のオブジェクトならば、正式OID804を保持することになる。

【0105】オブジェクト状態管理テーブル110のオブジェクト状態管理エントリ806は、オブジェクト108の状態を管理するために、以下の情報を保持している。

- 【0106】（1）「管理対象のオブジェクト識別子（OID）」
- （2）「オブジェクト状態フラグ」（生成／更新の状態であることを示す値を保持する。新規の場合には、フラグをONする）
- （3）「オブジェクトデータのアドレス」（管理対象オブジェクトのオブジェクトデータ領域の先頭アドレス；オブジェクト制御ヘッダ801の種別の欄を先頭アドレスとしている）
- （4）「部分更新情報のアドレス」（管理対象オブジェクトに対して要求された部分更新情報のアドレス）
- （5）「オブジェクト識別子（OID）変換情報のアドレス」（管理対象オブジェクトに対して要求されたOID変換情報のアドレス）
- （6）「オブジェクト識別子（OID）変換テーブル登録要請フラグ」
- （7）「更新チェーン409の続き要素エントリのアドレス」

次に、図9、10、11を用いて、図2のステップ205に示したデータベース管理システム（DBMS）クライアントにおけるオブジェクト間リンク設定APIの処理の詳細について説明する。図9は、本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト間リンク設定APIの処理手順を示すフローチャートである。

【0107】ステップ901において、オブジェクト管理部107は、ディクショナリ定義管理部111に定義されているオブジェクト間関連定義情報を参照する。対象となる関連定義は、アプリケーションプログラム105からリンク設定APIのパラメタとして関連識別子で指定され、この関連識別子に対応する関連定義情報を、ディクショナリ定義管理部111から取得する。

【0108】ステップ902において、オブジェクト管

理部107は、リンク元オブジェクトが新規で、かつ仮OIDが割り当てられていないかを判定する。新規であるか否かは、図8に示したオブジェクト状態管理エントリ806内の「状態フラグ」で判定する。新規の場合には、「状態フラグ」がONになっている。仮OIDが割り当てられているか否かは、図8に示したオブジェクト制御ヘッダ801内の「自分自身のOID」で、「OID形式」を参照することにより判定する。「OID形式」には、仮OID形式であることを示す値若しくは、正式OID形式であることを示す値が設定されているので、これらの値が設定されていない場合には、仮OIDが割り当てられていないと判定する。

【0109】仮OIDが割り当てられていない場合は、ステップ903に進み、仮OIDが割り当てられている場合は、ステップ906に進む。

【0110】ステップ903において、仮OID管理部112は、新規の仮OIDを発行する。このとき、仮OID管理部112は、生成通番113をインクリメントする。

【0111】次に、ステップ904において、オブジェクト管理部107は、リンク元オブジェクトのオブジェクト制御ヘッダ801およびオブジェクト状態管理エントリ806内に、発行した仮OIDを設定する。即ち、オブジェクト制御ヘッダ801の自分自身のOIDに、仮OID803を設定する。生成通番113には、ステップ903でインクリメントした生成通番を設定する。また、オブジェクト状態管理エントリ806の「管理対象OID」に、仮OIDを設定する。

【0112】次に、ステップ905において、オブジェクト管理部107は、オブジェクト状態管理エントリ806内の「OID変換テーブル登録要請フラグ」をONにする。このフラグの値に従って、データベース管理システム（DBMS）サーバ103で反映処理を行なうときに、OID変換テーブル125に、このオブジェクトについてのエントリを登録することにより、OID変換テーブル125へのエントリ登録を効率良く行なうことができる。ステップ905を終えた後、ステップ906に進む。

【0113】ステップ906からステップ909においては、リンク先オブジェクトについて、ステップ902からステップ905と同様して、オブジェクト識別子（OID）を判定し、仮OIDを割り付ける処理を行なう。

【0114】即ち、ステップ906において、オブジェクト管理部107は、リンク先オブジェクトが新規で、かつ仮OIDが割り当てられていないかを判定する。仮OIDが割り当てられていない場合は、ステップ907に進み、仮OIDが割り当てられている場合は、ステップ910に進む。

【0115】ステップ907において、仮OID管理部

112は、新規に仮OIDを発行する。このとき、仮OID管理部112は、生成通番113をインクリメントする。

【0116】次に、ステップ908において、リンク先オブジェクトのオブジェクト制御ヘッダ801およびオブジェクト状態管理エントリ806内に、発行した仮OIDを設定する。

【0117】次に、ステップ909において、オブジェクト管理部107は、オブジェクト状態管理エントリ806内の「OID変換テーブル登録要請フラグ」をONにする。ステップ909を終えた後、ステップ910へ進む。

【0118】ステップ910において、オブジェクト管理部107は、当該処理でリンク先オブジェクトに対して行なう更新を記録しておくための部分更新要求情報を作成する。この更新要求情報作成処理の詳細については、図10、11を用いて後述する。

【0119】さらに、ステップ911において、オブジェクト管理部107は、ステップ910と同様に、当該処理でリンク先オブジェクトに対して行なう更新を記録しておくための更新要求情報を作成する。この更新要求情報作成処理の詳細についても、図10、11を用いて後述する。

【0120】ステップ911を終えた後、リンク設定API処理を終了する。

【0121】次に、図10、11を用いて、図9のステップ910、911に示したデータベース管理システム(DBMS)クライアントにおける更新情報作成の処理の詳細について説明する。図10は、本発明の一実施形態によるオブジェクト管理方法におけるリンク設定時の更新情報作成の処理手順を示すフローチャートであり、図11は、本発明の一実施形態によるオブジェクト管理方法におけるリンク設定時の更新情報作成処理で扱われるデータ構造の説明図である。

【0122】る。

【0123】ステップ1001において、オブジェクト管理部107は、ディクショナリ定義管理無111から取得した関連定義情報を参照する。取得した関連定義情報から、オブジェクト内にリンク先情報を保持する位置のオフセットを取得し、リンク先情報をリンク先オブジェクト内に埋め込む。ここで、リンク先情報とは、リンク先オブジェクトのOIDである。

【0124】即ち、図11において、オブジェクト1101aをリンク先オブジェクトとし、オブジェクト1101bをリンク元オブジェクトとすると、リンク元オブジェクト110bのリンク先情報1102に、リンク先オブジェクト1101aの「仮OIDa」を埋め込む。

【0125】ステップ1002において、オブジェクト管理部107は、リンク元オブジェクトから、「オブジェクト制御ヘッダ」に保持しているアドレスを参照し

て、オブジェクト状態管理エントリを取得する。

【0126】即ち、図11において、リンク元オブジェクト1101bの「オブジェクト制御ヘッダ」の先頭に保持しているアドレスを参照して、オブジェクト状態管理テーブル110のオブジェクト状態管理エントリ1104bを取得する。

【0127】ステップ1003において、オブジェクト管理部107は、リンク先オブジェクトのOIDが仮OIDであるかを判定する。この判定は、OID内の「OID形式」を参照して行われる。仮OIDの場合は、ステップ1004に進み、仮OIDでない場合は、ステップ1006に進む。

【0128】即ち、図11において、リンク先オブジェクト1101aの「オブジェクト制御ヘッダ」内の「自分自身のOID」の「OID形式」(図8参照)を参照して判定する。ここでは、リンク先オブジェクト1101a及びリンク元オブジェクト1101bのいずれも仮OIDであるので、ステップ1004に進むものとする。

【0129】ステップ1004において、オブジェクト管理部107は、リンク先オブジェクトのOIDを埋め込む位置(リンク元オブジェクト内のリンク先情報)のオフセットをもとに、OID変換情報を作成する。このOID変換情報に設定するオフセットは、データベース管理システム(DBMS)サーバ103が、仮OIDが埋め込まれた位置を効率良く得るために用いられる。

【0130】即ち、図11において、OID変換情報1105には、更新位置オフセットが設定される。この更新位置オフセットは、リンク元オブジェクト1101bのリンク先情報11102のオフセットである。

【0131】次に、ステップ1005において、オブジェクト管理部107は、リンク元オブジェクトのオブジェクト管理管理エントリに、OID変換情報のアドレスを設定する。1つのオブジェクトに対して複数の仮OIDが埋め込まれる場合に対処するため、OID変換情報をアドレスポインタで連結し、オブジェクト状態管理エントリには、OID変換情報のチェーンの先頭の要素(OID変換情報1105)のアドレスを保持する。

【0132】即ち、図11において、リンク元オブジェクト1101bのオブジェクト管理管理エントリ1104bに、OID変換情報1105のアドレスを設定する。アドレスの設定位置は、図8において説明したように、OID変換要請フラグの前の位置である。複数の仮OIDが埋め込まれる場合には、図11に示すように、次のOID変換情報を、前のOID変換情報1105のアドレスポインタで連結するようにする。

【0133】ステップ1006において、オブジェクト管理部107は、リンク先情報を設定するための部分更新情報を作成する。部分更新情報は、「更新位置オフセット」、「更新範囲」および「更新データ内容の先頭ア

ドレス」から構成されている。1つのオブジェクトに対する複数の部分更新に対処するため、部分更新情報は、「アドレスポインタ」で連結する。オブジェクト状態管理エントリには、部分更新情報チェーンの先頭要素のアドレスを保持する。

【0134】即ち、図11において、部分更新情報1103は、「更新位置オフセット」、「更新範囲サイズ」、「更新データ内容の先頭アドレス」及び「次の部分更新情報のアドレスポインタ」から構成されている。

「更新データ内容の先頭アドレス」は、リンク先情報1102の先頭アドレスである。オブジェクト状態管理エントリ1104bは、部分更新情報のチェーンの先頭の要素（部分更新情報1103）のアドレスを保持している。その位置は、図8において説明したように、OID変換情報1105のアドレスの前の位置である。

【0135】ステップ1007において、オブジェクト管理部107は、リンク元オブジェクトのオブジェクト状態管理エントリに部分更新情報のアドレスを設定する。既に、部分更新情報が作成されている場合には、新たに作成した部分更新情報を、既作成の部分更新情報のチェーンに連結する。

【0136】即ち、図11において、オブジェクト状態管理エントリ1104bの「OID変換要請フラグ」の2つ前の位置に、部分更新情報1103のアドレスを設定する。

【0137】ステップ1008において、オブジェクト管理部107は、リンク元オブジェクト状態管理エントリを設定する。即ち、リンク元オブジェクト状態管理エントリの「状態フラグ」の「更新」を示す値をONにする。このフラグは、後のトランザクションコミットで反映情報電文を作成するときに、当該オブジェクトに部分更新の反映が要求されていること判定するために用いる。

【0138】即ち、図11において、リンク元オブジェクト1101bのオブジェクト状態管理エントリ1104bのオブジェクト状態フラグの「更新」を示す値をONにする。

【0139】ステップ1009において、オブジェクト管理部107は、リンク元オブジェクト状態管理エントリを、反映要求チェーンに連結する。

【0140】即ち、図11において、リンク元オブジェクト1101bのオブジェクト状態管理エントリ1104bを、反映要求チェーン409に連結する。

【0141】ステップ1009を終えた後、更新情報作成処理を終了する。

【0142】ここで、図11において、図10において説明した更新情報作成処理を実行した結果、リンク元オブジェクト1101bのオブジェクトデータ領域内にあるリンク先情報1102に、リンク先オブジェクト1101aに割り付けられたOID（仮OIDa）が設定さ

れる。この設定要求を記録しておくために、部分更新情報1103が割り付けられる。部分更新情報1103は、オブジェクト状態管理エントリ1104bの部分更新情報チェーンにつながる。

【0143】また、リンク先オブジェクト1101aのOIDが仮OIDの場合は、OID変換情報1105が作成され、オブジェクト状態管理エントリ1104bのOID変換情報チェーンにつながる。リンク先が正式OIDを持つ場合は、このリンク設定に関しては、OID変換情報は不要となる。

【0144】なお、上述した例では、双方向関連として、リンク先オブジェクトについても、同様の構造をとっている。

【0145】次に、図12、図13を用いて、図2のステップ206に示したデータベース管理システム（DBMS）クライアントにおけるトランザクションコミットするAPIの処理の詳細について説明する。図12は、本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミットAPIの処理手順を示すフローチャートであり、図13は、本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミット要求の通信電文データ構造の説明図である。

【0146】ステップ1201において、データベース管理システム（DBMS）クライアント106は、オブジェクト制御ヘッダからオブジェクト状態管理エントリの反映要求チェーンの先頭要素アドレスを取得する。

【0147】即ち、図11において、オブジェクト1101bのオブジェクト制御ヘッダからオブジェクト状態管理エントリ1104bを取得し、オブジェクト状態管理エントリ1104bから反映要求チェーン409の先頭アドレスを取得する。

【0148】ステップ1202において、データベース管理システム（DBMS）クライアント106は、ステップ1201で取得した反映要求チェーンに要素があるか、すなわち、反映要求がなされているかを判定する。反映要求がなされていない場合は、ステップ1203に進み、反映要求がなされている場合は、ステップ1205に進む。

【0149】ここで、リンク先オブジェクト1101aとリンク元オブジェクト1101bがリンクされているので、反映要求がなされているとして、ステップ1205に進んで説明する。

【0150】ステップ1205において、データベース管理システム（DBMS）クライアント106は、仮OID管理部の生成通番を、OID変換総数として、通信電文に設定する。この値は、サーバでOID変換が必要な総数を伝え、OID変換テーブルの領域を初期化するために用いられる。

【0151】ここでは、オブジェクト1101a、1101bの2つのオブジェクトに対して仮OIDが発行さ

れており、この2つのオブジェクトを一括送信する場合について考えると、図13に示したコミット要求の通信電文1301の変換OID総数302には、“2”が設定される。

【0152】ステップ1206において、データベース管理システム（DBMS）クライアント106は、反映要求チェーン409をたどり、反映が必要なオブジェクト状態管理エントリ1104bのアドレスを取得する。

【0153】ステップ1207において、データベース管理システム（DBMS）クライアント106は、ステップ1206で取得したオブジェクト状態管理エントリ内の「状態フラグ」で「新規」を示す値がONであるかを判定する。「新規フラグ」がONの場合には、ステップ1208に進み、オブジェクト生成要求電文を作成する。オブジェクト生成要求電文は、図13のコマンドパラメタ1303に示されるように作成されるが、このオブジェクト生成要求電文作成の詳細については、図14を用いて後述する。「新規フラグ」がONでない場合には、ステップ1209に進む。

【0154】ステップ1209において、データベース管理システム（DBMS）クライアント106は、反映要求チェーンの続きがあるかを判定する。続きがある場合には、ステップ1206に戻り、ステップ1206からステップ1209までの処理を繰り返すことにより、すべての生成要求についての電文が作成される。2回目のステップ1208において、図13のコマンドパラメタ1304に示されるオブジェクト生成要求電文が作成される。ステップ1209で続きがない場合は、ステップ1210に進む。

【0155】ステップ1210において、データベース管理システム（DBMS）クライアント106は、反映要求チェーンの先頭に戻る。即ち、図11に示したオブジェクト制御ヘッダ1101bを参照して、オブジェクト状態管理テーブル110のオブジェクト状態管理エントリ1104bのアドレスを取得する。

【0156】ステップ1211において、データベース管理システム（DBMS）クライアント106は、反映要求チェーンをたどり、反映が必要なオブジェクト状態管理エントリのアドレスを取得する。即ち、図11に示した反映要求チェーン409をたどり、次のオブジェクト状態管理エントリのアドレスを取得する。

【0157】ステップ1212において、データベース管理システム（DBMS）クライアント106は、オブジェクト状態管理エントリ内の状態フラグで更新を示す値がONであるかを判定する。更新フラグがONの場合、ステップ1213に進み、オブジェクト更新要求電文を作成する。オブジェクト更新要求電文は、図13のコマンドパラメタ1305に示されるように作成されるが、このオブジェクト更新要求電文作成の詳細については、図15を用いて後述する。更新フラグがONでない

場合、ステップ1214に進む。

【0158】ステップ1214において、データベース管理システム（DBMS）クライアント106は、反映要求チェーンの続きがあるかを判定する。続きがある場合は、ステップ1211に戻り、ステップ1211からステップ1214までの処理を繰り返すことにより、すべての更新要求についての電文が作成される。2回目のステップ1213において、図13のコマンドパラメタ1308に示されるオブジェクト更新要求電文が作成される。ステップ1214で続きがない場合は、ステップ1215に進む。これにより、すべての更新要求についての電文が作成される。

【0159】ステップ1215において、データベース管理システム（DBMS）クライアント106は、トランザクションコミット要求の電文を作成する。データベース管理システム（DBMS）クライアント106は、通信管理部114にコミット要求の電文領域を確保し、図13に示すトランザクションコミット要求電文のコマンドパラメタブロック1311に、コミット要求であることを示すコマンド識別子を設定する。なお、コマンド識別子の後方の次ブロックオフセットには、次ブロックがあるときには、その次ブロックのオフセットが設定される。

【0160】次に、ステップ1216において、データベース管理システム（DBMS）クライアント106は、通信管理部114より、サーバ103へ電文を送信し、トランザクションコミット要求処理を終了する。

【0161】なお、ステップ1202において、反映要求がないとして、ステップ1203に進んだ場合に、ステップ1203では、ステップ1215と同様にして、トランザクションコミット要求の電文を作成し、さらに、ステップ1204において、データベース管理システム（DBMS）クライアント106は、通信管理部114より、サーバ103へ電文を送信し、トランザクションコミット要求処理を終了する。

【0162】次に、図13を用いて、トランザクションコミット要求電文のデータ構造について説明する。通信電文1301の電文制御ヘッダは、（1）「電文の形式」（サーバへの送信データであることを示す値など）、（2）「電文のサイズ」、（3）「先頭ブロックオフセット」（先頭のコマンドパラメタブロック1303の位置を示すオフセット値；通信電文1301の先頭からのオフセット値として与えられている）、（4）「ブロック総数」（通信電文1301に保持するコマンドパラメタブロック1303の総数）、（5）「変換OID総数」（図12のステップ1205で設定）から構成されている。「ブロック総数」に設定されたデータから、通信電文1301が5つのコマンドパラメタブロック1303、1304、1305、1308、1311によって構成されていることが示されている。また、

「変換OID総数1302」により、この通信電文処理で変換OIDテーブルに登録する要素の数が“2”であることが示されている。

【0163】1番目のコマンドパラメタブロック1303には、「コマンド識別子」に“生成”が設定されており、「オブジェクトデータ」として“オブジェクトa”が設定されており、オブジェクトaの生成要求が記されている。コマンドパラメタブロック1303では、「OID変換テーブル登録要請フラグ」がONに設定されており、データベース管理システム（DBMS）サーバ103で要求を処理するときに、OID変換テーブル125にこのオブジェクトについてのエントリを登録するように指示されている。また、「次ブロックオフセット」により、コマンドパラメタブロック1304の先頭位置を指示している。

【0164】2番目のコマンドパラメタブロック1304には、コマンドパラメタブロック1303と同様に、オブジェクトbについての生成要求が記されている。コマンドパラメタブロック1304の「OID変換テーブル登録要請フラグ」がONに設定されており、また、「次ブロックオフセット」により、コマンドパラメタブロック1305の先頭位置を指示している。

【0165】3番目のコマンドパラメタブロック1305には、「コマンド識別子」に“更新”が設定されており、「更新対象データ」として“仮OIDa”が設定されており、オブジェクトaの更新要求が記されている。

「次ブロックオフセット」により、コマンドパラメタブロック1308の先頭位置を指示している。「部分更新数」には、図11に示した部分更新情報1103のチェインの数が設定される。「部分更新情報オフセット」は、部分更新情報1306の先頭位置を指示している。

「OID変換数」は、図11に示したOID変換情報1105のチェインの数が設定される。「OID変換情報オフセット」は、OID変換情報1307の先頭位置を指示している。

【0166】さらに、部分更新情報1306には、オブジェクトbへのリンク先情報の設定情報が記されている。「部分更新オフセット」、「部分更新オフセット」及び「更新データ内容」（リンク先情報）は、図11に示した部分更新情報1103の内容が設定される。

【0167】また、OID変換情報1307には、リンク先情報内のオブジェクトbの仮OIDを変換するための情報が記されている。「OID変換位置オフセット」は、図11に示したOID変換情報1105の内容が設定される。

【0168】4番目のコマンドパラメタブロック1308、部分更新情報1309、OID変換情報1310には、コマンドパラメタブロック1305、部分更新情報1306、OID変換情報1307と同様にして、オブジェクトbの更新要求が記されている。

【0169】5番目のコマンドパラメタブロック1311には、コミット要求が記されている。このコマンドパラメタブロック1311により、データベース管理システム（DBMS）サーバ103でトランザクションコミットが行なわれることにより、コマンドパラメタブロック1303からコマンドパラメタブロック1310による要求が、データストアへ反映されることが保証される。

【0170】次に、図14を用いて、図12のステップ1208に示したデータベース管理システム（DBMS）クライアントにおけるオブジェクト生成要求電文作成の処理の詳細について説明する。図14は、本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト生成要求電文作成の処理手順を示すフローチャートである。

【0171】ステップ1401において、データベース管理システム（DBMS）クライアント106は、オブジェクト108のデータサイズをもとに電文に必要な領域サイズを算出する。

【0172】ステップ1402において、通信管理部114は、ステップ1401で算出した値を元に、電文領域を確保する。

【0173】ステップ1403において、データベース管理システム（DBMS）クライアント106は、オブジェクト状態管理エントリ1104b内の「OID変換テーブル登録要請フラグ」がONであるかを判定する。

「OID変換テーブル登録要請フラグ」がONの場合は、ステップ1404に進み、ONでない場合には、ステップ1405に進む。

【0174】ステップ1404において、データベース管理システム（DBMS）クライアント106は、コマンドパラメタ中の「OID変換テーブル登録要請フラグ」をONに設定する。即ち、図13に示したコマンドパラメタブロック1303、1304中の「OID変換テーブル登録要請フラグ」をONに設定する。

【0175】ステップ1405において、データベース管理システム（DBMS）クライアント106は、オブジェクト108のデータをコマンドパラメタ中の新規オブジェクトデータ領域にコピーする。即ち、オブジェクトデータをコマンドパラメタブロック1303、1304の中の「オブジェクトデータ」の領域に設定する。ステップ1405を終えた後、生成要求電文作成の処理を終了する。

【0176】次に、図15を用いて、図12のステップ1213に示したデータベース管理システム（DBMS）クライアントにおけるオブジェクト更新要求電文作成の処理の詳細について説明する。図15は、本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト更新要求電文作成の処理手順を示すフローチャートである。

【0177】ステップ1501において、データベース管理システム(DBMS)クライアント106は、オブジェクト状態管理エントリの部分更新情報のチェーンをたどり、更新要求数をカウントし、電文のコマンドパラメタブロックに必要なサイズを算出する。即ち、図11に示した部分更新情報1103のチェーンをたどり、部分更新情報のエントリ数をカウントする。

【0178】ステップ1502において、データベース管理システム(DBMS)クライアント106は、OID変換情報チェーンをたどり、変換する仮OID数をカウントし、電文のコマンドパラメタブロックに必要なサイズを算出する。即ち、図11に示したOID変換情報1105のチェーンをたどり、OIDの変換要求数をカウントする。

【0179】ステップ1503において、データベース管理システム(DBMS)クライアント106は、ステップ1501およびステップ1502で得た値をもとに、更新要求コマンドパラメタブロック全体に必要なサイズを算出し、通信管理部114は、電文領域を確保する。

【0180】ステップ1504において、データベース管理システム(DBMS)クライアント106は、更新対象オブジェクトのOIDを通信電文に設定する。即ち、図11に示したオブジェクト状態管理エントリ1104bの仮OIDbを、図13に示したコマンドパラメタブロック1305の「更新対象」に設定する。

【0181】ステップ1505において、データベース管理システム(DBMS)クライアント106は、部分更新情報1103のチェーンをたどり、1つの部分更新情報1103を取得する。

【0182】ステップ1506において、データベース管理システム(DBMS)クライアント106は、ステップ1505で得た部分更新情報を参照し、「部分更新位置オフセット」、「更新範囲」を取得し、これを通信電文に設定する。また、オブジェクトキャッシュ上のオブジェクトの更新内容を参照し、更新データ内容を電文に設定する。即ち、図11に示した部分更新情報1103の「部分更新位置オフセット」、「部分更新範囲サイズ」を取得し、図13に示した部分更新情報1306の「部分更新位置オフセット」、「部分更新範囲サイズ」に設定し、また、更新データ内容(リンク先情報)を設定する。

【0183】ステップ1507において、データベース管理システム(DBMS)クライアント106では、部分更新情報チェーンで続き要素があるかを判定する。続きがある場合は、ステップ1505に戻り、ステップ1505からステップ1507の処理を繰り返し、すべての部分更新情報の電文を作成する。即ち、図13に示した部分更新情報1306及び部分更新情報1309が作成される。

【0184】一方、ステップ1507の判定において、続きがない場合は、ステップ1508に進む。

【0185】ステップ1508において、データベース管理システム(DBMS)クライアント106は、更新情報の数および電文中で更新情報を設定した位置のオフセット情報を、コマンドパラメタブロックのブロック制御ヘッダに設定する。即ち、図13に示したコマンドパラメタブロック1305の「部分更新数」に、更新情報の数を設定し、「部分更新情報オフセット」に、電文中で更新情報を設定した位置のオフセット情報を設定する。

【0186】ステップ1509において、データベース管理システム(DBMS)クライアント106は、更新データ中にOID変換情報があるか否かを判定する。即ち、図11に示したオブジェクト状態管理エントリ1104bの「OID変換情報アドレス」が、OID変換情報1105の先頭アドレスを保持しているか判定する。このオブジェクト内に、OID変換が必要であるかを判定する。

【0187】OID変換情報がある場合は、ステップ1510に進み、OID変換情報がない場合は、ステップ1512に進む。

【0188】ステップ1510において、データベース管理システム(DBMS)クライアント106は、OID変換情報チェーンをたどり、1つのOID変換情報を得る。即ち、図11に示したOID変換情報1105を取得する。

【0189】次に、ステップ1511において、データベース管理システム(DBMS)クライアント106は、OID変換情報を電文に設定する。ここで、OID変換する位置を示すオフセットを設定する。

【0190】即ち、図13に示したOID変換情報1307を設定する。また、このとき、「次OID変換オフセット」を設定する。

【0191】ステップ1512において、データベース管理システム(DBMS)クライアント106は、OID変換情報チェーンで続き要素があるかを判定する。続きがある場合は、ステップ1510に戻り、ステップ1510からステップ1512の処理を繰り返す。これにより、すべてのOID変換情報の電文が作成される。即ち、OID変換情報1307、1310が設定される。続きがない場合は、ステップ1513に進む。

【0192】ステップ1513において、データベース管理システム(DBMS)クライアント106は、ステップ1509からステップ1512の処理で作成したOID変換情報の数、および電文中でのOID変換情報の位置を示すオフセットをコマンドパラメタブロックのブロック制御ヘッダに設定する。即ち、図13に示したコマンドパラメタブロック1305、1308の「OID変換数」に、OID変換情報の数を設定し、「OID変

換情報オフセット」に、電文中でのOID変換情報の位置を示すオフセットを設定する。

【0193】ステップ1513を終えた後、当該更新要求電文作成処理を終了する。

【0194】次に、図16を用いて、図2のステップ206に示したデータベース管理システム(DBMS)サーバにおけるトランザクションコミットの処理の詳細について説明する。図16は、本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミットの処理手順を示すフローチャートである。

【0195】ステップ1601において、サーバアプリケーション118aは、通信管理部107から、電文のアドレスを取得する。

【0196】ステップ1602において、124は、通信電文中の変換OID総数(図13に示した変換OID総数1302)をもとに、総数分の要素配列を保持可能な領域を確保し、OID変換テーブル125を初期化する。

【0197】ステップ1603において、サーバアプリケーション118は、ステップ1601で得た通信電文から、電文制御ヘッダの「先頭ブロックオフセット」を参照して、コマンドパラメタブロック1303を取得する。

【0198】ステップ1604において、サーバアプリケーション118は、コマンドパラメタブロック1303の「コマンド識別子」が「生成」を示す値であるかを判定する。「生成」の場合は、ステップ1605に進む。ステップ1605において、オブジェクト管理部119は、生成要求の電文解析処理を行なう。生成要求の電文解析処理の詳細については、図17を用いて後述する。ステップ1605を終えた後、ステップ1610に進む。

【0199】ステップ1610において、次のコマンドパラメタブロックを取得する。図13に示すように、コマンドパラメタブロック1303の「コマンド識別子」は「生成」であり、続くコマンドパラメタブロック1304の「コマンド識別子」も「生成」であるので、ステップ1605による生成コマンド処理を実行する。

【0200】一方、ステップ1604において、「生成」でない場合は、ステップ1606へ行き、コマンド識別子が「更新」を示す値であるかを判定する。

【0201】図13に示したコマンドパラメタブロック1305、1308の「コマンド識別子」は「更新」であるので、ステップ1607に進み、オブジェクト管理部119は、更新要求の電文解析処理を行なう。更新要求の電文解析処理の詳細については、図18を用いて後述する。ステップ1607を終えた後、ステップ1610に進む。

【0202】一方、ステップ1606で「更新」でない場合は、ステップ1608へ行き、コマンド識別子が

「コミット」を示す値であるかを判定する。

【0203】図13に示したコマンドパラメタブロック1311の「コマンド識別子」は「コミット」であるので、ステップ1609に進み、サーバアプリケーション118は、トランザクション管理部117へトランザクションコミットの要求を行なう。ステップ1609を終えた後、当該コミット処理を終了する。

【0204】一方、ステップ1608において、要求種別が「コミット」でない場合は、ステップ1610に進み、ステップ1610では、電文中の次のコマンドパラメタブロックを取得し、ステップ1604に戻る。以上の繰り返しにより、「コミット」要求までのすべての「生成」および「更新」の反映要求に対する処理が行なわれる。

【0205】次に、図17、図19を用いて、図16のステップ1605に示したデータベース管理システム(DBMS)サーバにおけるオブジェクト生成コマンド処理の詳細について説明する。図17は、本発明の一実施形態によるオブジェクト管理方法における生成コマンドの処理手順を示すフローチャートであり、図19は、本発明の一実施形態によるオブジェクト管理方法におけるDBMSサーバ内でのオブジェクトデータの構造の説明図である。

【0206】ステップ1701において、オブジェクト管理部119は、コマンドパラメタブロック中のオブジェクトデータサイズを取得し、オブジェクトデータサイズのオブジェクトを保持できるように、オブジェクトキャッシュ領域を確保する。

【0207】即ち、図13に示したコマンドパラメタブロック1303、1304の「オブジェクトデータ」の構成は、図8に示したような構成になっており、図8に示したオブジェクトデータ108のオブジェクト制御ヘッダ801の中の「オブジェクトの全体のサイズ」からオブジェクトデータサイズを取得する。

【0208】ステップ1702において、オブジェクト管理部119は、コマンドパラメタブロック中のオブジェクトデータを、ステップ1701で確保した領域にコピーする。即ち、図13に示したコマンドパラメタブロック1303、1304の「オブジェクトデータ」を、図19に示したオブジェクトキャッシュ120のオブジェクト1901aにコピーする。

【0209】ステップ1703において、オブジェクト管理部119は、格納制御部126にオブジェクト格納を要求する。格納位置が確定することにより、永続識別子(PID)が与えられ、OID管理部124よりこのオブジェクトに対して正式OIDが割り付けられる。ここで、割り付けられた正式OIDをオブジェクトキャッシュ120上のオブジェクト1901aの制御ヘッダの自分自身のOIDの保持域(図19における正式OIDaの領域)に設定する。即ち、図19のオブジェクトキ



キャッシュ120上のオブジェクト1901aの「自分自身のOID」に「正式OIDa」が設定され、オブジェクト1901bの「自分自身のOID」に「正式OIDb」が設定される。

【0210】ステップ1704において、オブジェクト管理部119は、このオブジェクトデータに対して、オブジェクト状態管理エントリを割り付け、初期化する。これにより、このオブジェクトが活性化状態となる。

【0211】ステップ1705において、オブジェクト管理部119は、コマンドパラメタブロック1303、1304中の「OID変換テーブル登録要請フラグ」がONであるかを判定する。フラグがONの場合は、ステップ1706に進む。

【0212】ステップ1706において、オブジェクト管理部119は、オブジェクトの仮OID内の通番を元に、OID変換テーブルに登録する。即ち、図13に示したコマンドパラメタブロック1303、1304中の構造は、図8に示したとおりであり、図8に示したオブジェクト108のオブジェクト制御ヘッダ801の中の「自分自身のOID」の中の仮OID803内の生成通番をもとに、図19に示したOID変換テーブル125において、このオブジェクトに関するエントリの登録位置を確定し、活性化されているオブジェクトデータのアドレスを設定する。

【0213】ステップ1706を終えた後、または、ステップ1705でフラグがONでない場合は、生成コマンド処理を終了する。

【0214】次に、図18を用いて、図16のステップ1607に示したデータベース管理システム(DBMS)サーバにおけるオブジェクト更新コマンド処理の詳細について説明する。図18は、本発明の一実施形態によるオブジェクト管理方法における更新コマンドの処理手順を示すフローチャートである。

【0215】ステップ1801において、オブジェクト管理部119は、更新対象オブジェクトをオブジェクトキャッシュに活性化する。ここで、更新対象が新規オブジェクトの場合は、先に、図17で説明した処理により、すでに活性化状態になっており、OID変換テーブルを参照することにより、対象オブジェクトを取得することができる。また、更新対象がデータストアに格納されている既存のオブジェクトの場合は、通信電文中に指定される正式OIDにより、周知の活性化機構を用いて、データストアから活性化されるものとする。

【0216】ステップ1802において、オブジェクト管理部119は、ブロック制御ヘッダを参照して、ブロック中の1つの部分更新情報を取得する。即ち、図13に示したコマンドパラメタブロック1305の「部分更新情報オフセット」から部分更新情報1306を取得する。

【0217】ステップ1803において、オブジェクト

管理部119は、ステップ1802で取得した部分更新情報をもとに、ステップ1801で取得したオブジェクトデータを書き換える。即ち、ステップ1801において、初期化した領域に部分更新情報を入れる。

【0218】ステップ1804において、オブジェクト管理部119は、更新情報のブロック制御ヘッダを参照し、先のステップ1803で処理した部分更新情報に続きがあるか判定する。

【0219】続きがある場合は、ステップ1802に戻り、すべての部分更新が行なわれるようにする。即ち、図13に示したコマンドパラメタブロック1308の「部分更新情報オフセット」から部分更新情報1309を取得し、取得した部分更新情報をもとに、ステップ1801で取得したオブジェクトデータを書き換える。続きがない場合は、ステップ1805に進み。

【0220】ステップ1805において、オブジェクト管理部119は、ブロック制御ヘッダを参照し、ブロック中にOID変換情報を保持しているか判定する。即ち、図13に示したコマンドパラメタブロック1305の「OID変換情報オフセット」からOID変換情報を保持しているか判定する。保持していない場合は、ステップ1812に進み、保持している場合は、ステップ1806に進む。

【0221】ステップ1806において、オブジェクト管理部119は、ブロック中から1つのOID変換情報を取得する。即ち、図13に示したOID変換情報1307を取得する。

【0222】ステップ1807において、オブジェクト管理部119は、ステップ1806で取得したOID変換情報内のオフセットをもとに、オブジェクトデータ内に埋め込まれている仮OIDを取得する。即ち、図13に示したコマンドパラメタブロック1305の中の「更新対象」の「仮OIDa」を取得する。

【0223】ステップ1808において、オブジェクト管理部119は、ステップ1807で取得した仮OIDの内容を参照し、仮OIDの内部データに保持されている生成通番を取得する。即ち、仮OIDの構造は、図8に示したようであり、図8の仮OID803の「生成通番」を取得する。

【0224】ステップ1809において、オブジェクト管理部119は、ステップ1808で得られた値をもとに、この値を、OID変換テーブル配列の要素位置として、オブジェクトに対応するエントリを取得する。このエントリには、図19に示したように、オブジェクトキャッシュ120上に活性化されたオブジェクトのアドレスが設定されている。オブジェクト1901の制御ヘッダには、自分自身の正式OIDが設定されているので、これを参照することにより、正式OIDを取得する。

【0225】ステップ1810において、オブジェクト管理部119は、オブジェクトデータ内の仮OIDを、

ステップ1809で得られた正式OIDに書き換える。即ち、図19に示したオブジェクト1901aの中で、図8に示したオブジェクト制御ヘッダ801の中の「リンク先OID」に正式OIDbを書き込む。

【0226】ステップ1811において、オブジェクト管理部119は、ブロック中に続きのOID変換情報があるか判定する。OID変換情報がある場合は、ステップ1806に戻り、すべてのOID変換を処理するまで、ステップ1806からステップ1811の処理を繰り返す。これにより、図19に示したオブジェクト1901bの中の「リンク先OID」に正式OIDaを書き込まれる。

【0227】一方、ステップ1811において、OID変換情報がない場合は、ステップ1812に進む。

【0228】ステップ1812において、オブジェクト管理部119は、格納制御部126に対して、これまでの更新をデータストアに反映するよう要求する。ステップ1812を終えた後、当該更新コマンド処理を終了する。

【0229】ここで、図19を用いて、データベース管理システム(DBMS)サーバにおけるオブジェクトデータ構造について説明する。

【0230】オブジェクトa1901aとオブジェクトb1901bが、オブジェクトキャッシュ120上に活性化されている。OID管理部124のOID変換テーブル125には、オブジェクトaとオブジェクトbのそれぞれのエントリが登録されている。ここで、オブジェクトaは、生成通番でa番目に生成されたものとし、生成順に従って、OID変換テーブル125の配列でa番目にエントリが登録されている。同様に、オブジェクトbについても、OID変換テーブル125の配列でb番目にエントリが登録されている。

【0231】図17のステップ1703により、オブジェクトa1901aとオブジェクトb1901bのオブジェクト制御ヘッダで、「自分自身のOID」の保持領域に、それぞれ、「正式OIDa」、「正式OIDb」が設定されている。

【0232】また、図18のステップ1810により、オブジェクトa1901aとオブジェクトb1901bのリンク先情報で、「リンク先OID」に、それぞれ、「正式OIDb」、「正式OIDa」が設定されている。

【0233】以上、本発明の一実施形態について説明したが、これは例示したものであり、これによって本発明が制限されるものではない。

【0234】なお、上述した例において、オブジェクト間の関連の多重度は1対1としていたが、多対多の関連についても、本発明を適用することができる。多対多関連の実装方法としては、一般に、直にリンク先OIDを保持するのではなく、OIDの集合を管理するコレクシ

ョン機能を介して実装され、コレクションの要素のOIDがリンク先オブジェクトのOIDであると解釈される。この場合、基本的にはOIDを扱うので、上記の方式で仮OIDを正式OIDに変換すれば、実装することが可能である。

【0235】また、上述した例において、生成通番の一意性を保証する期間単位をトランザクションとしているが、サーバと通信を行なうタイミングや、データベース(DB)接続、クライアントプロセス単位など、クライアントアプリケーションの形態に応じて適当に単位を設定することも考えられる。

【0236】本実施形態によれば、クライアントーサーバ構成のオブジェクト指向データベースシステムにおいて、新規オブジェクトに対して生成順通番をもとにした仮OIDを用いるにより、正式OIDを必要とするようなクライアントアプリケーションでのオブジェクト操作を、データベース管理システム(DBMS)サーバとの通信を行なうことなく(通信回数を削減して)、効率良く処理することが可能となる。

【0237】また、複数の仮OIDを一括してクライアントからサーバに送信することにより、通信回数を削減することができる。

【0238】また、反映要求の電文にOID変換情報を付加しておくことにより、仮OIDから正式OIDへの変換を効率よく行えるようになる。

【0239】

【発明の効果】本発明によれば、クライアントーサーバ型で構成されるオブジェクト指向データベースシステムにおいて、システム内でのオブジェクト識別性を保証しつつ、サーバと通信を行うことなく、クライアントアプリケーションでのオブジェクト操作を効率よく処理できる。

【図面の簡単な説明】

【図1】本発明の一実施形態によるオブジェクト管理方法を適用したオブジェクト指向データベース(OODB)システムの構成図である。

【図2】本発明の一実施形態によるオブジェクト管理方法におけるクライアントアプリケーションでのアプリケーションプログラムの処理フローの概念を示すフローチャートである。

【図3】本発明の一実施形態によるオブジェクト管理方法におけるデータベース(DB)接続APIの処理手順を示すフローチャートである。

【図4】本発明の一実施形態によるオブジェクト管理方法において用いるデータベース管理システム(DBMS)クライアントで取り扱う制御データの構造の説明図である。

【図5】本発明の一実施形態によるオブジェクト管理方法においてクライアントアプリケーションが作成する通信電文のデータ構造の説明図である。

【図6】本発明の一実施形態によるオブジェクト管理方法におけるトランザクション開始API関数コールの処理手順を示すフローチャートである。

【図7】本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト生成APIの処理手順を示すフローチャートであり、

【図8】図7に示したオブジェクト生成APIの処理によって生成されるデータベース管理システム(DBMS)クライアントのオブジェクトデータ構造の説明図である。

【図9】本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト間リンク設定APIの処理手順を示すフローチャートである。

【図10】本発明の一実施形態によるオブジェクト管理方法におけるリンク設定時の更新情報作成の処理手順を示すフローチャートである。

【図11】本発明の一実施形態によるオブジェクト管理方法におけるリンク設定時の更新情報作成処理で扱われるデータ構造の説明図である。

【図12】本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミットAPIの処理手順を示すフローチャートである。

【図13】本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミット要求の通信電文データ構造の説明図である。

【図14】本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト生成要求電文作成の処理手順を示すフローチャートである。

【図15】本発明の一実施形態によるオブジェクト管理方法におけるオブジェクト更新要求電文作成の処理手順を示すフローチャートである。

【図16】本発明の一実施形態によるオブジェクト管理方法におけるトランザクションコミットの処理手順を示すフローチャートである。

【図17】本発明の一実施形態によるオブジェクト管理方法における生成コマンドの処理手順を示すフローチャ\*

\*ートである。

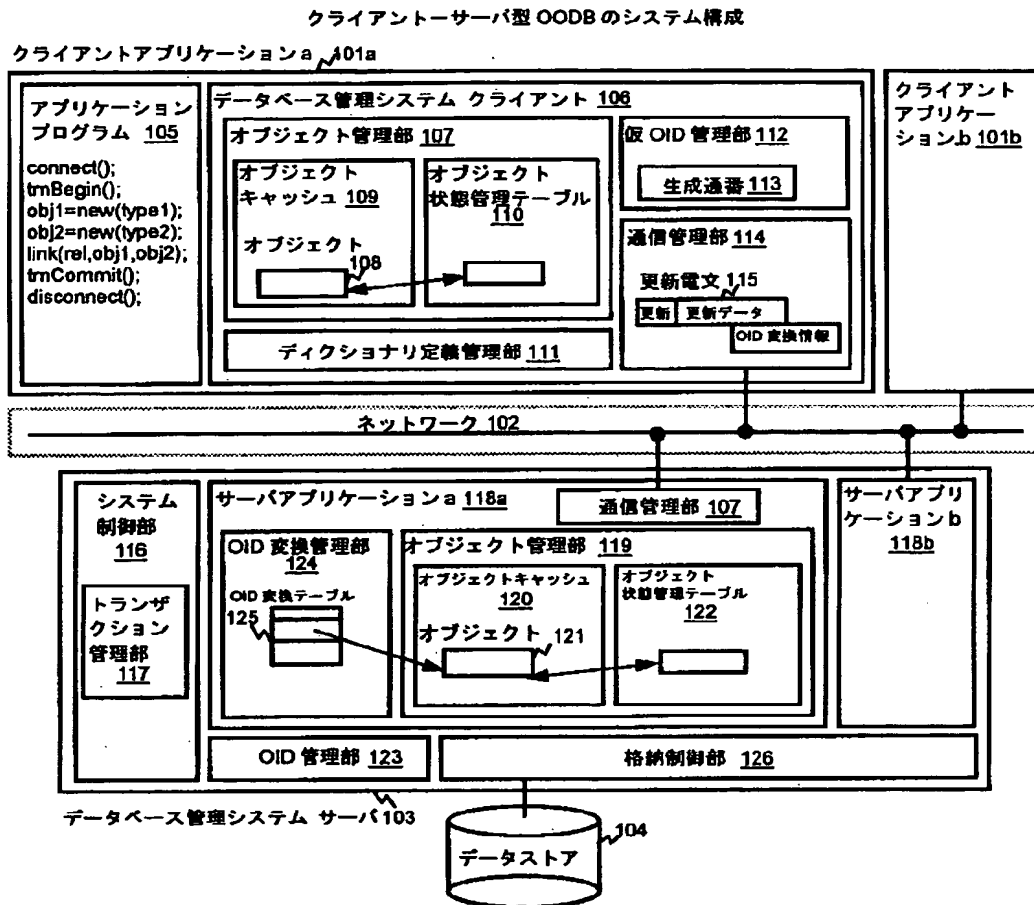
【図18】本発明の一実施形態によるオブジェクト管理方法における更新コマンドの処理手順を示すフローチャートである。

【図19】本発明の一実施形態によるオブジェクト管理方法におけるDBMSサーバ内でのオブジェクトデータの構造の説明図である。

#### 【符号の説明】

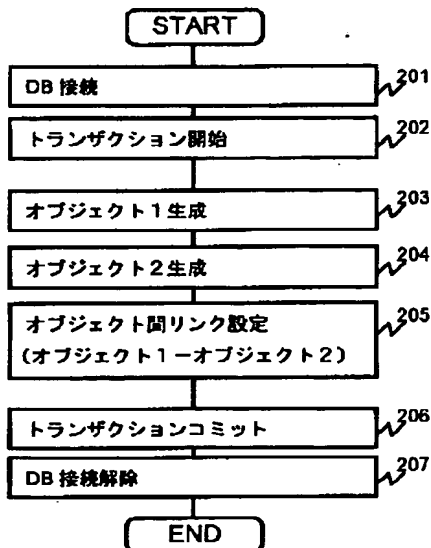
- 101 a…クライアントアプリケーション a
- 101 b…クライアントアプリケーション b
- 102…ネットワーク
- 103…データベース管理システム サーバ
- 104…データストア
- 105…アプリケーションプログラム
- 106…データベース管理システムクライアント
- 107…オブジェクト管理部
- 108…オブジェクト
- 109…オブジェクトキャッシュ
- 110…オブジェクト状態管理テーブル
- 111…ディクショナリ定義管理部
- 112…仮OID管理部
- 113…生成通番
- 114…通信管理部
- 115…更新電文
- 116…システム制御部
- 117…トランザクション管理部
- 118 a…サーバアプリケーション a
- 118 b…サーバアプリケーション b
- 119…オブジェクト管理部
- 120…オブジェクトキャッシュ
- 121…オブジェクト
- 122…オブジェクト状態管理テーブル
- 123…OID管理部
- 124…OID変換管理部
- 125…OID変換テーブル
- 126…格納制御部

【図1】



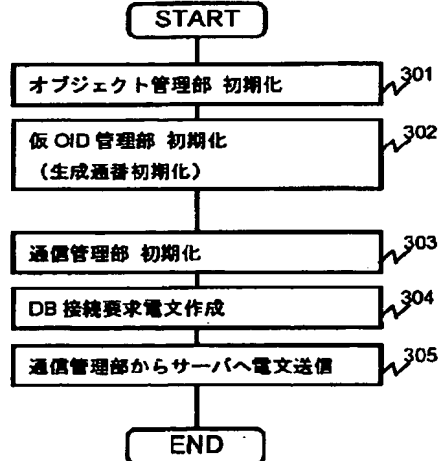
【図2】

アプリケーションプログラムの処理フロー



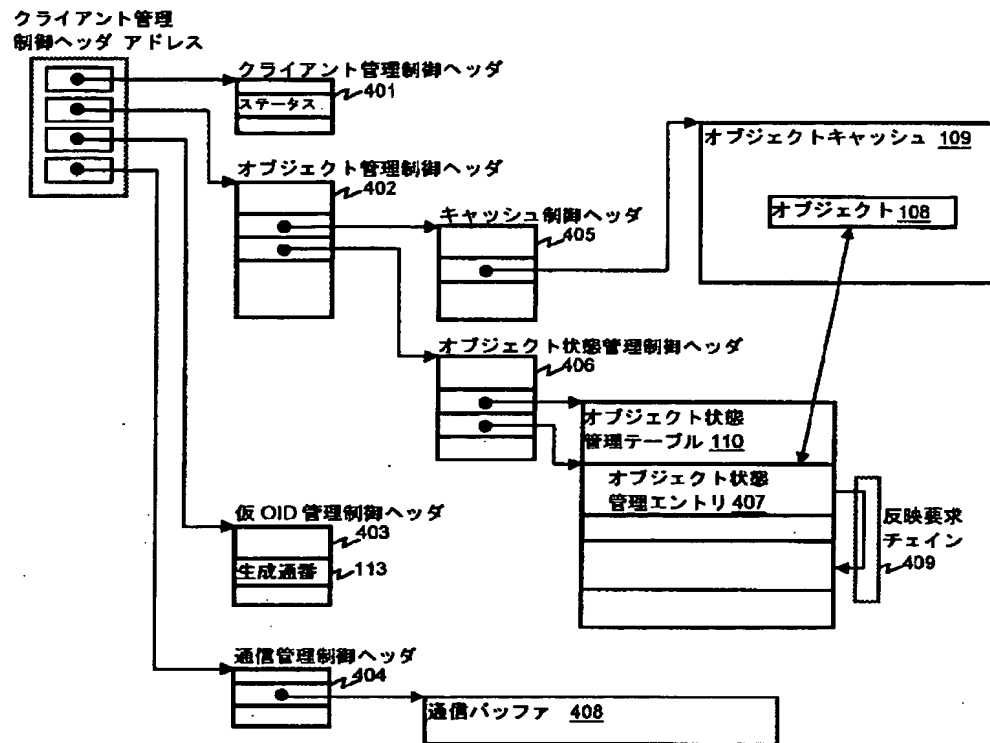
【図3】

DBMS クライアント DB 接続API の処理フロー



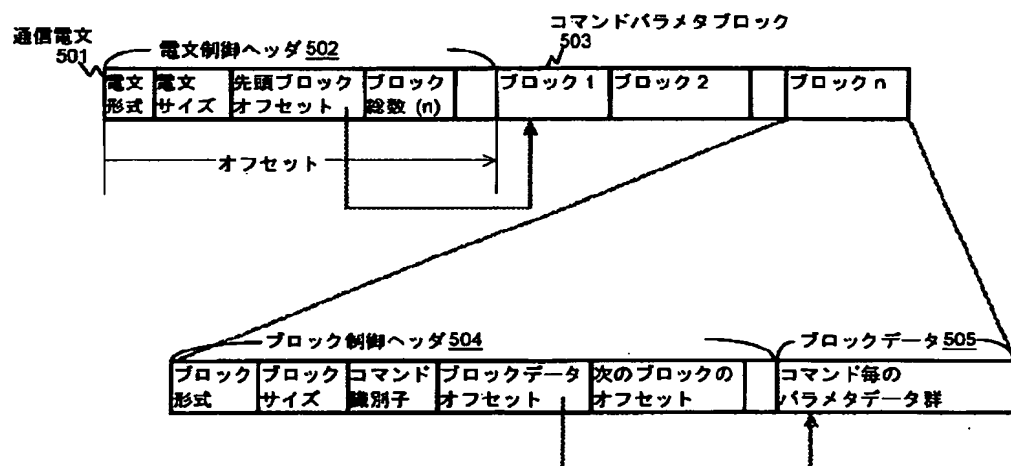
【図 4】

## DBMS クライアントの制御データ構造



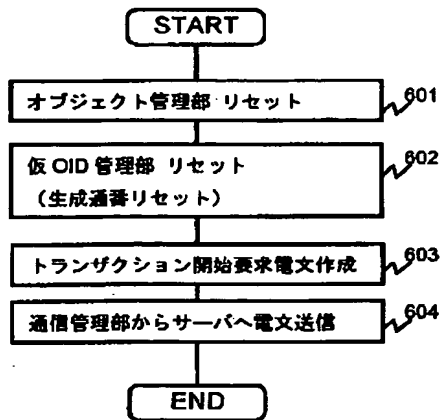
【図 5】

## 通信電文データ構造



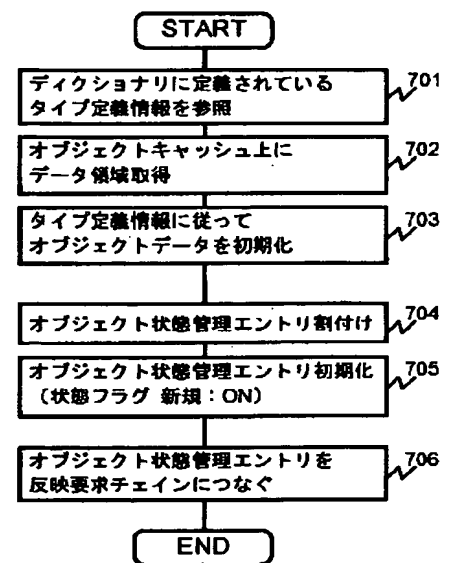
【図 6】

DBMS クライアント トランザクション開始 API の処理フロー



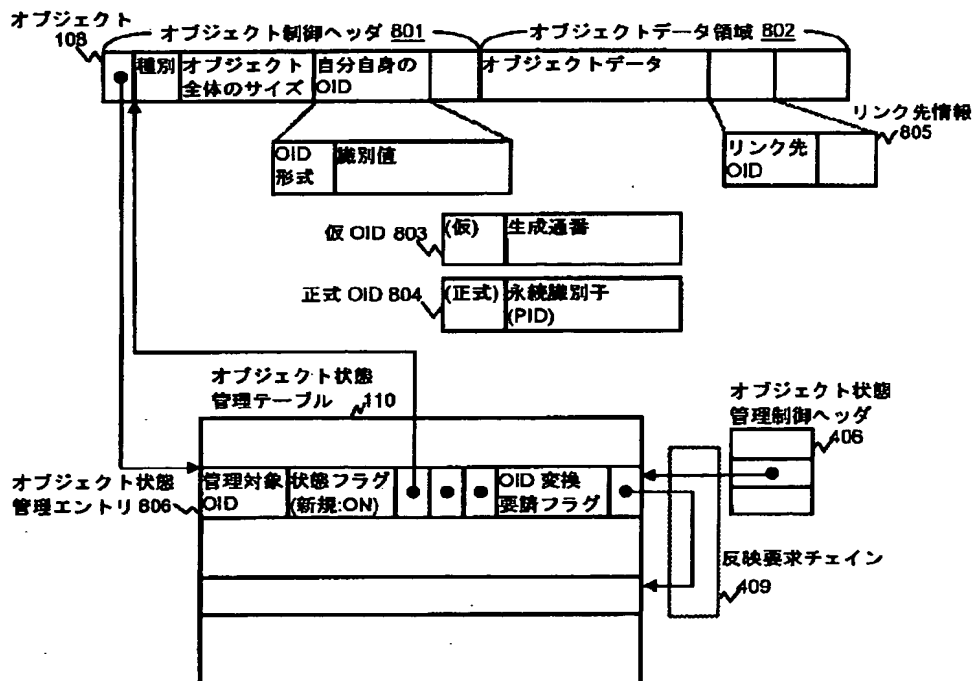
【図 7】

DBMS クライアント オブジェクト生成 API の処理フロー

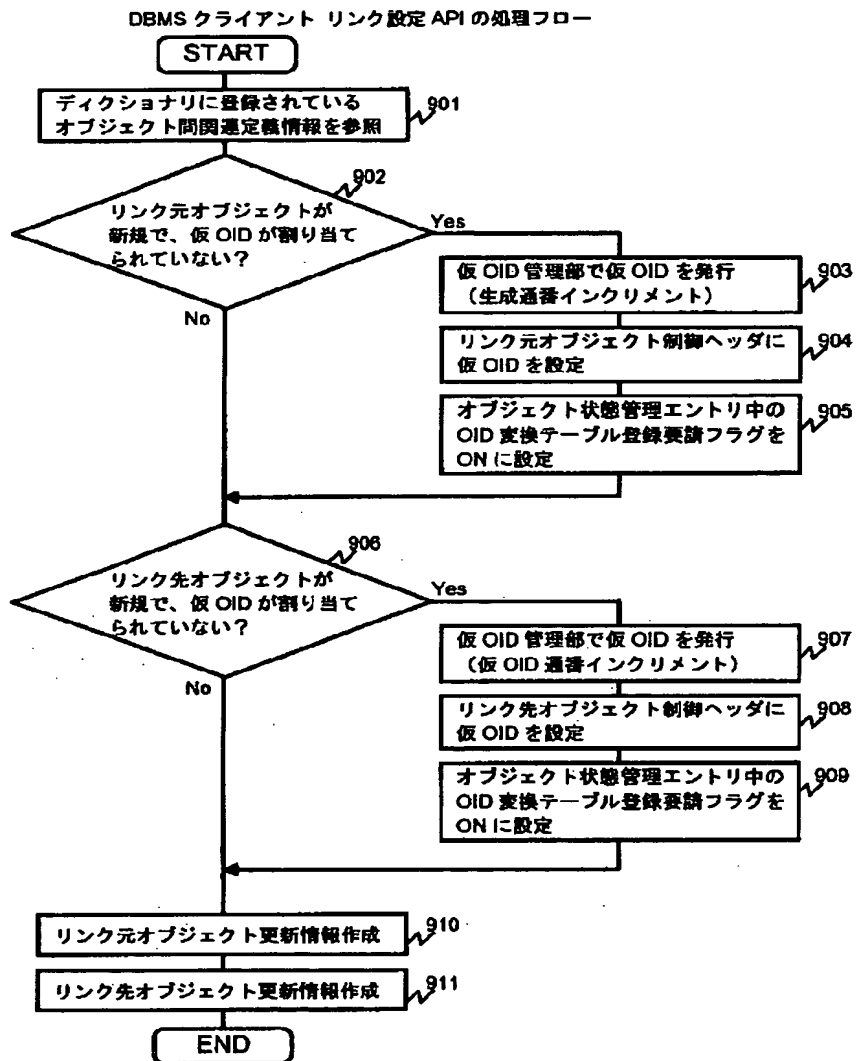


【図 8】

DBMS クライアントのオブジェクトデータ構造

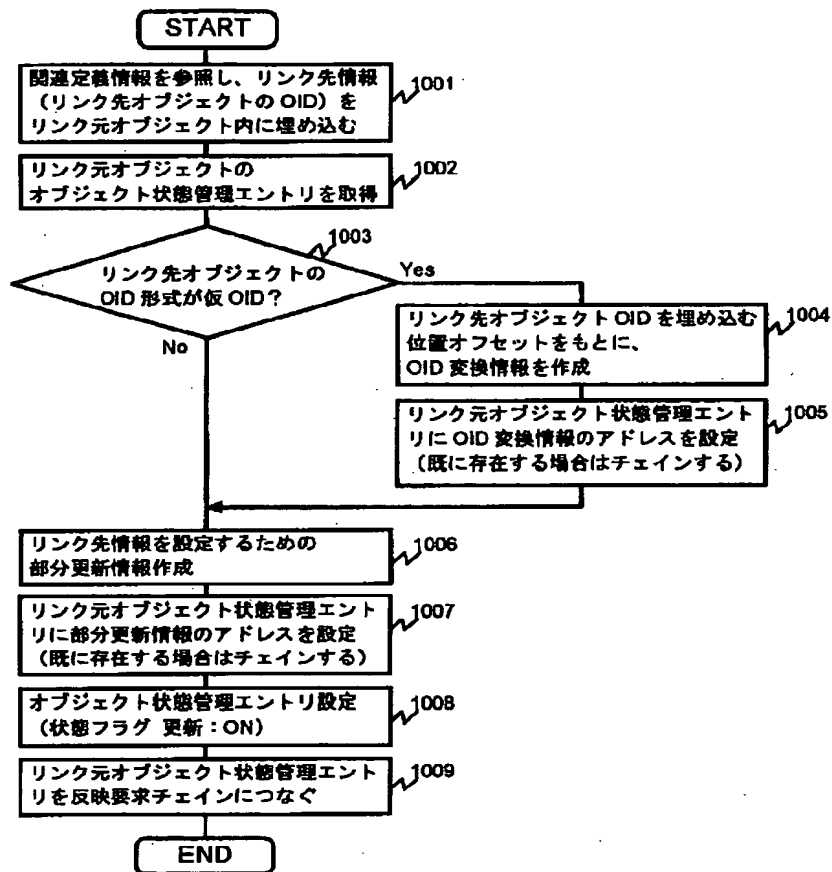


【図9】



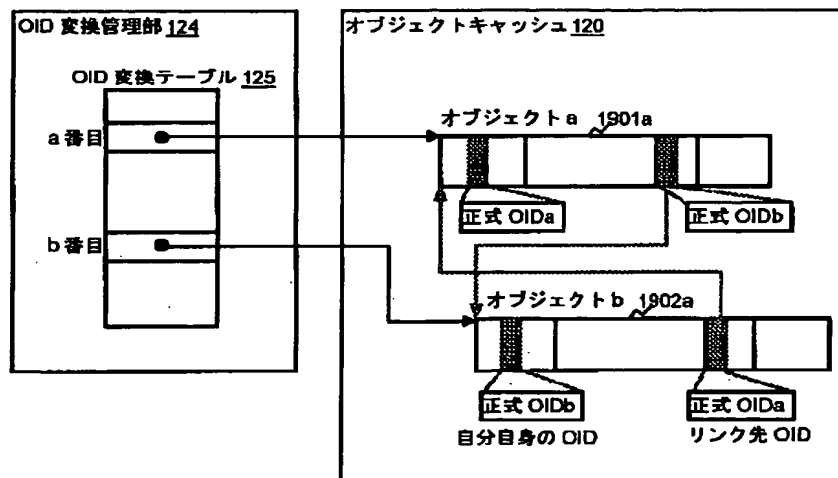
【図10】

DBMS クライアント リンク設定時 更新情報作成の処理フロー



【図19】

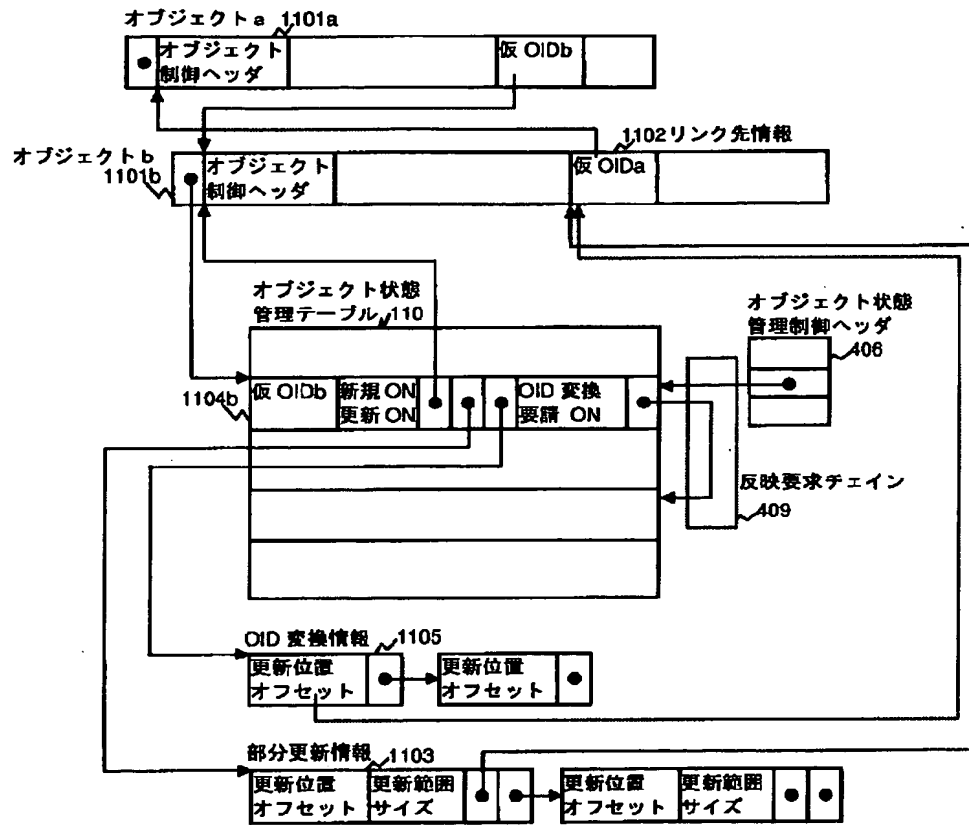
DBMS サーバのオブジェクトデータ構造





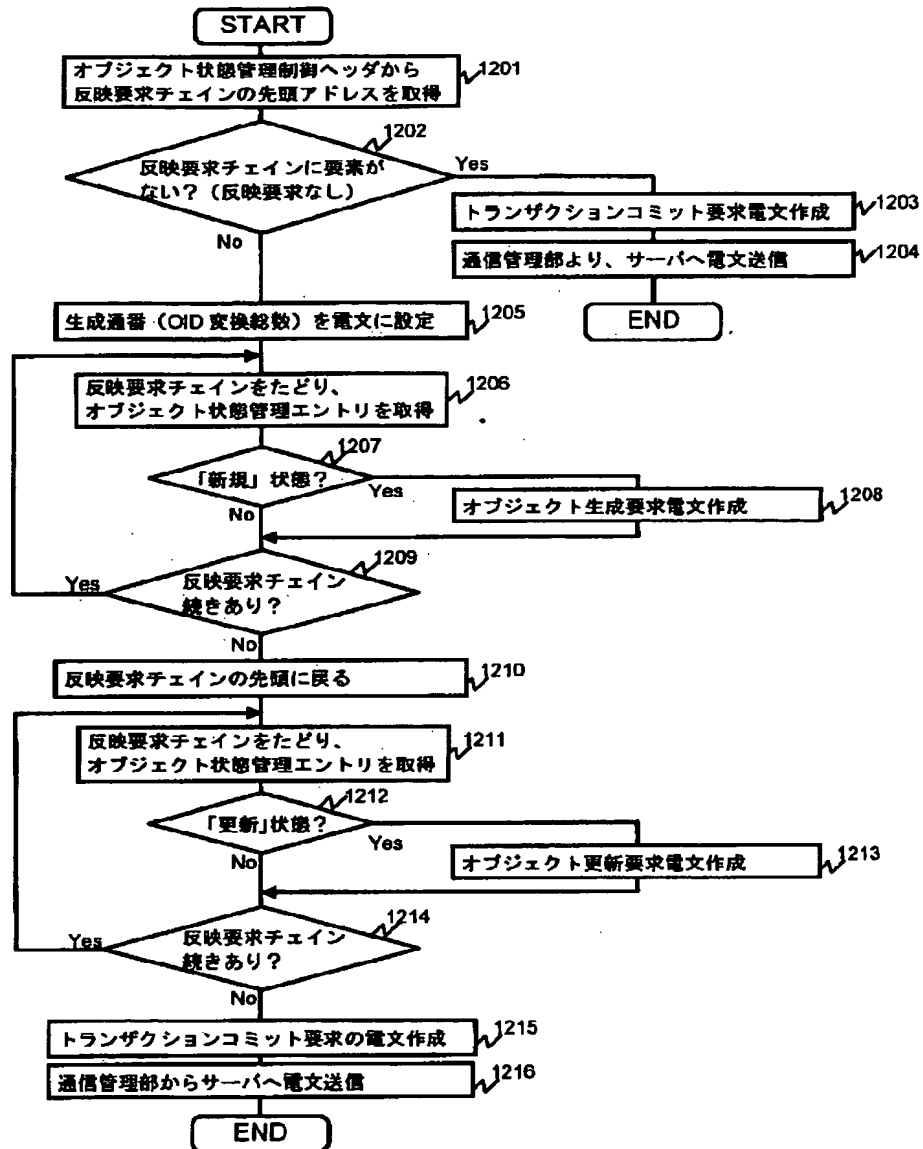
【図11】

## DBMS クライアント リンク設定のデータ構造



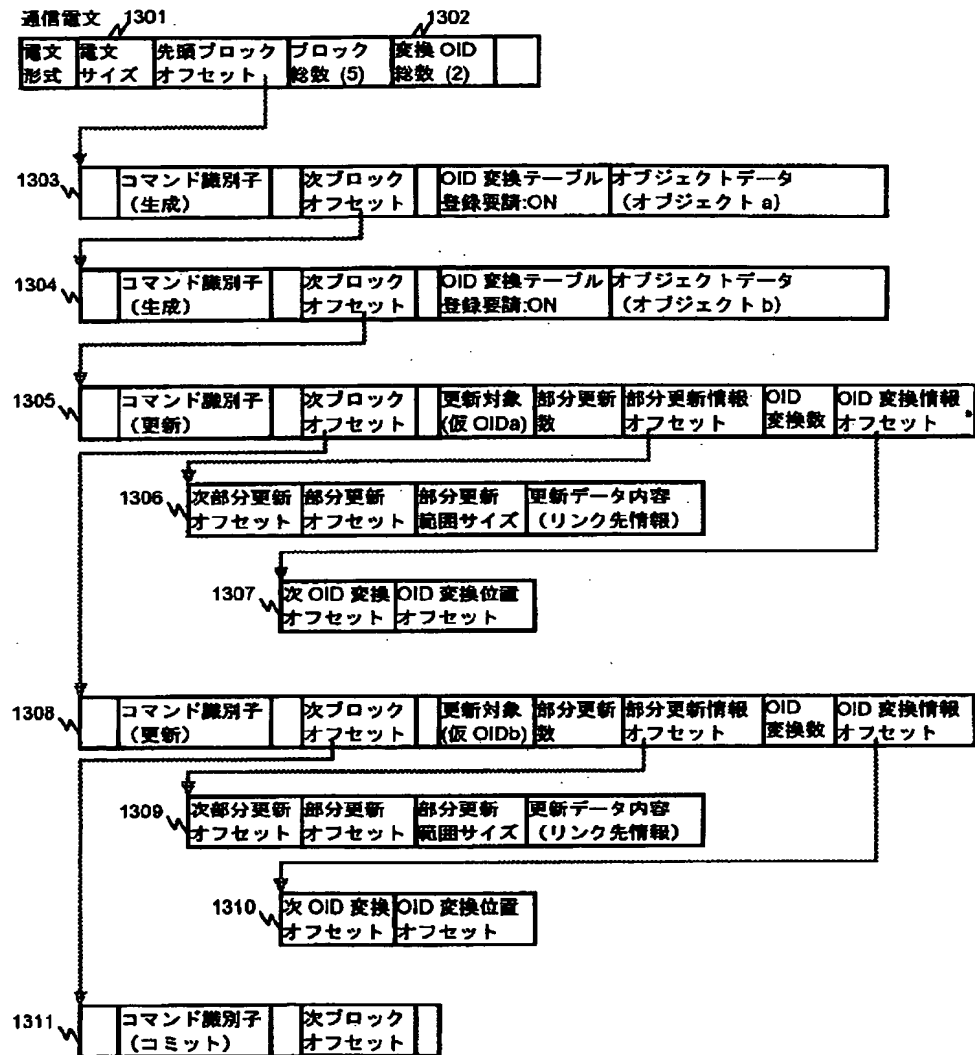
【図12】

## DBMSクライアント トランザクションコミットAPIの処理フロー



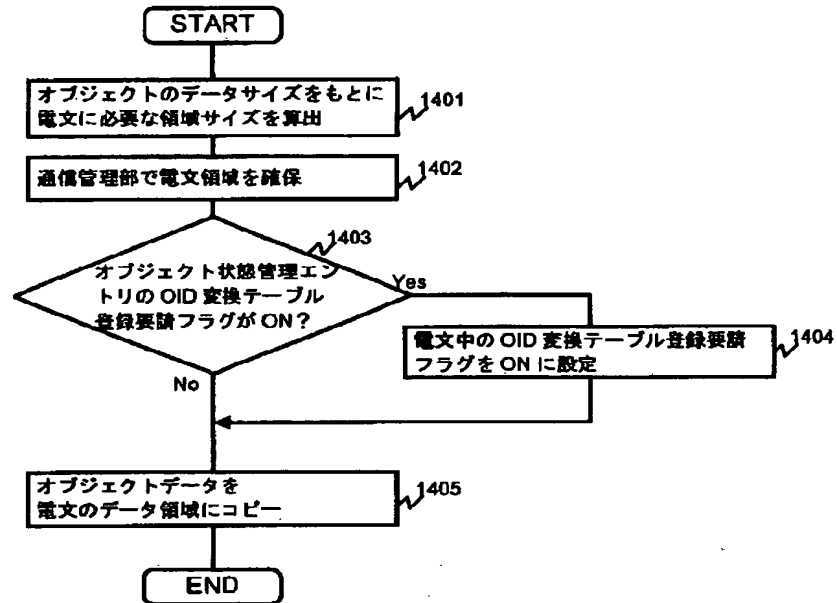
【図 13】

## DBMS クライアント トランザクションコミット要求電文データ構造



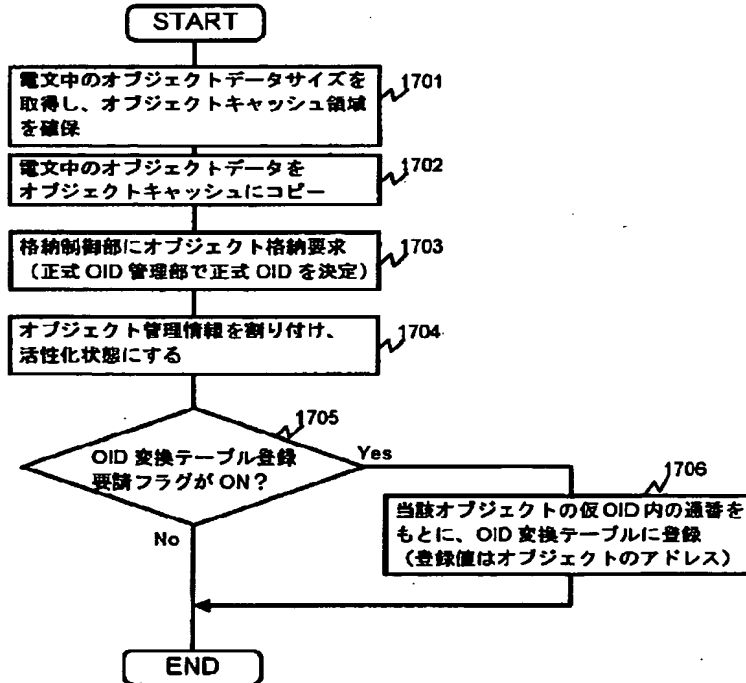
【図14】

## DBMS クライアント オブジェクト生成要求電文作成の処理フロー



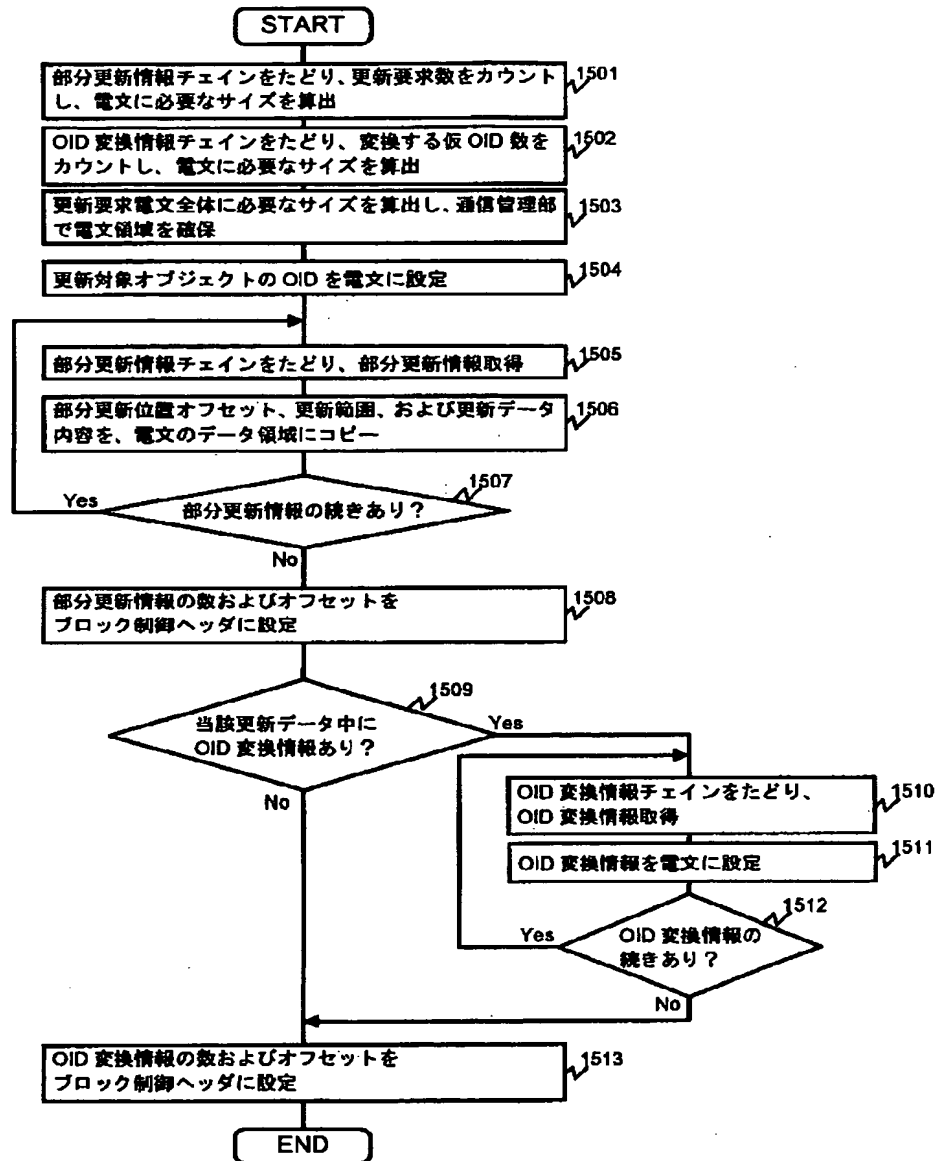
【図17】

## DBMS サーバ オブジェクト生成コマンド処理フロー



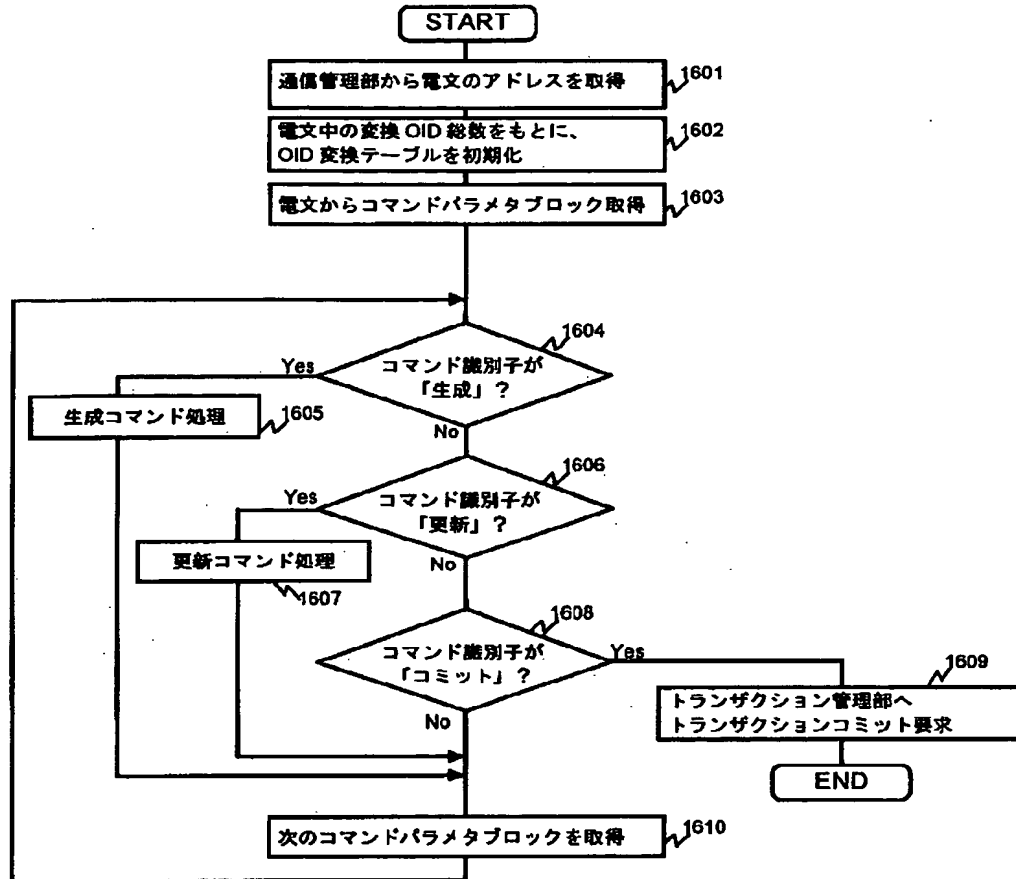
【図 15】

## DBMS クライアント オブジェクト更新要求電文作成の処理フロー



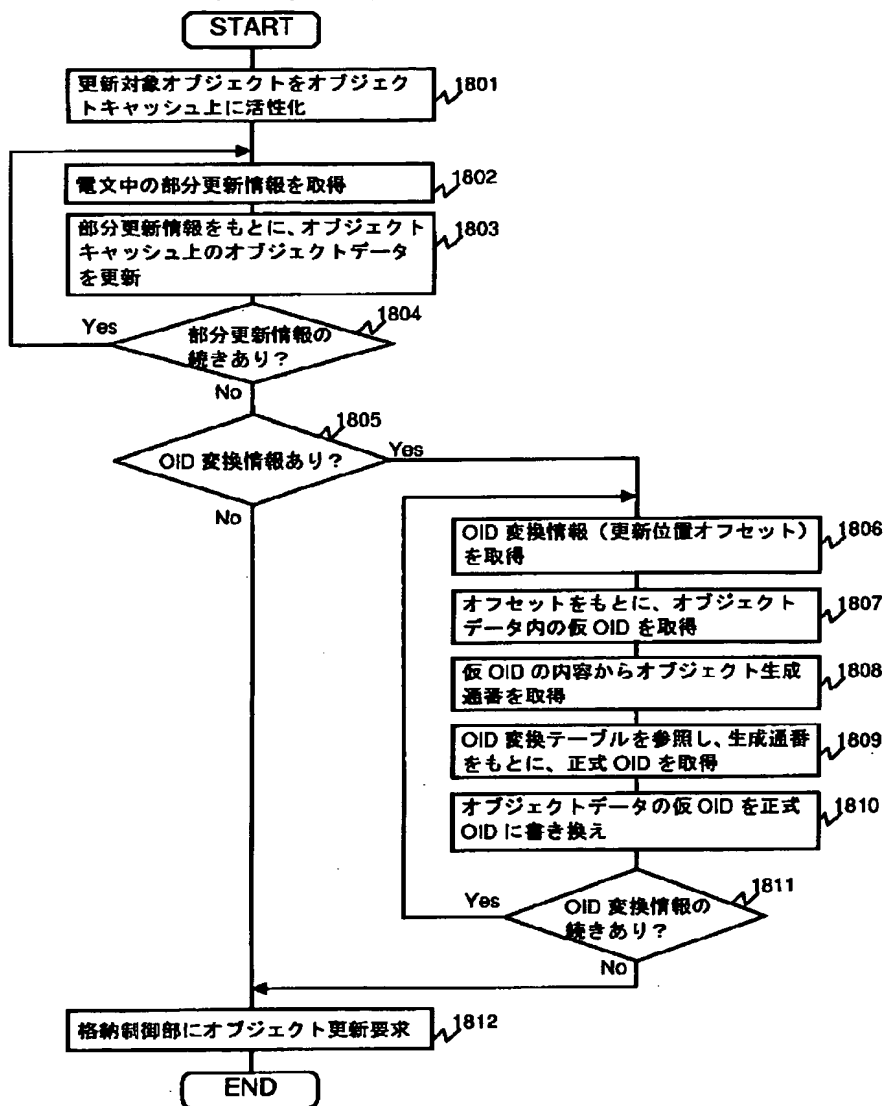
【図 16】

DBMS サーバ トランザクションコミットの処理フロー



【図 18】

## DBMS サーバ オブジェクト更新コマンド処理フロー



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☒ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

---

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**